



الجمهورية الجزائرية الديمقراطية الشعبية  
وزارة التعليم العالي و البحث العلمي  
المدرسة العليا لمساعدة التعليم التكنولوجي  
- سكيكدة -



التخصص: هندسة كهربائية

قسم التكنولوجيا

مذكرة التخرج لنيل شهادة أستاذ التعليم الثانوي  
بعنوان

دراسة وانجاز بطاقة إلكترونية للتعرف على الصور باستعمال  
ESP32-CAM

من المحاك:

مباركي خديجة

مرورة بن فرج

تحت إشراف الأستاذ:

الدكتور مريان ابراهيم

لجنة المناقشة:

- |                    |        |                |                       |
|--------------------|--------|----------------|-----------------------|
| م.ع.أ.ت.ت. سكيكدة. | رئيسا  | أستاذ محاضر-أ- | الدكتور مريان إبراهيم |
| م.ع.أ.ت.ت. سكيكدة. | مشرفا  | أستاذ محاضر-ب- | الدكتور تيفوتي عصام   |
| م.ع.أ.ت.ت. سكيكدة. | ممتحنا | أستاذ محاضر-أ- | الدكتور بوزناد خالد   |

السنة الجامعية 2025/2024

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

# شكر و عرفان

قبل أن يُسند الفضل لأهله، فإن أول من يُحمد وأعظم من يُشكر هو ربنا عز وجل

فالحمد لله واسع النعم، عظيم اللطف، جزيل العطاء، غامر الهبات.

الحمد لله الذي علم بالقلم، علم الإنسان ما لم يعلم، وأكرمنا بنور العلم بعد ظلمة الجهل.

وبعد، فقد قال رسول الله ﷺ: "من لا يشكر الناس لا يشكر الله"

نتوجه بجزيل الشكر وخالص الامتنان إلى الأستاذ تيفوتي عصام،

الذي رافقنا بنصائحه وتوجيهاته، فكان لعطائه العلمي والمعنوي بالغ الأثر في إتمام هذا العمل

فله منا كل التقدير والدعاء، ونسأل الله أن يبارك له في علمه وعمره وجهده

كما لا يفوتنا أن نشكر الأستاذ مريان إبراهيم وكل أعضاء لجنة المناقشة الكرام،

على تواضعهم وتفضلهم بقراءة وتصحيح هذا العمل، جزاكم الله عنا خير الجزاء.

## الإهداء

إلى سندي في كل درب، الى الذي له فضل في كل تفاصيل وصولي

إلى من أحمل اسمه بكل نخر

إلى أبي "العياشي بن فرج"

أهديك هذا العمل وفاء لتعبك، وامتنانا لا يفويه الكلام

إلى أمي "وروة"

إلى التي كلفت وضحت وأرشدتني إلى نعيم الدنيا

إلى أختي التي شد بها عضدي "جهاد"

إلى اخوتي زكرياء، اسلام، سميرة، سامية والزهرة

إلى أمينة ولؤي

إلى من كانت ملجئي وملاذي "لهام" والى "هينتي" وإلى رفيقات الدرب ساجدة، بشرى ورانية

ثم إلى بناتي وريحانات قلبي وإلى كل عزيز

مروة بن فرج

## الإهداء

إلى جدي الراحل

عسى أن نلتقي في الجنة فتنتهي الغربة وتطيب الجراح.

إلى مجاهدي قافلة الصمود

إلى غزة الجريحة

إلى المدافعين عن شرف الأمة

إلى أطفالها وطلابها الذين رحلوا قبل أن تكتمل الحكاية

سلاما على أرواحكم الطاهرة.

مروة بن فرج

## الإهداء

بسم الله الرحمان الرحيم

" وكان فضل الله عليك عظيما "

ماسلكنا البدايات إلا بتيسيره

ومابلغنا النهايات إلا بتوفيقه

وماحققنا الغايات إلا بفضله

الحمد لله حتى يبلغ الحمد منتهاه

بكل حب أهدي هذا النجاح...

إلى الذي زين اسمي بأجمل الألقاب

إلى الغالي الذي حملت اسمه فخرا

إلى من كلَّه الله بالهيبه والوقار

إلى من حصد الأشواك عن دربي وزرع لي الراحة بدلا منها

إلى أبي حبيبي "جمعي مباركي"

إلى من تحملت العناء في سبيل أن أكون كما أنا اليوم

إلى ملاكي في الحياة

إلى الجسر الصاعد بي إلى الجنة

إلى بسمة حياتي وسر وجودي

إلى أمي قرة عيني "حياة توماري"

إلى أبي الذي رسمني وأمي التي لونتني

إلى رفيق الخطوة الأولى والخطوة ما قبل الأخيرة

إلى من كان في سنوات العجاف سحابا ممطرا

إلى أخي وحبيبي "سمير"

إلى من جمعنا معهم بيت واحد  
إلى من كانوا خير سند  
إلى إخوتي خالد ونور الدين وأختي نوال  
إلى من كانت هدية الله لي  
إلى من أمسكت بيدي وانتشلتني من التيه إلى الطمأنينة  
إلى من وهبتني جهد أيامها حتى ختمت كلام ربي  
إلى نور حياتي "نور الهدى بن فردي"  
إلى من تفاءلت بالخير فوجدتها  
إلى من أهدتني إياها مدينة سكيكة  
إلى أمي الثانية "فوزية ديب"  
إلى ضلعي الثابت الذي لا يميل  
إلى ملاذي الأول والأخير  
إلى أختي وليدة الجامعة "شيماء حلايلي"  
إلى من تسعد عيني برؤية وجوههن  
إلى من يفرح فؤادي بسماع رنات ضحكتهن  
إلى من أحبوني فأحببت نفسي أكثر  
إلى "حلقة النور وصديقاتي"  
إلى "آل مشروع حافظة الذكر"  
إلى "آل الإتحاد العام الطلابي الحر"  
إلى "طلبة غزة الجامعين"  
إلى بناتي "إيناس وملوكي"  
إلى كل من اتسع قلبي لهم وضاعت هذه الوقة لذكرهم

غريمة الوحي والقلم

مباركي غريمة

## ملخص

هذا المشروع يتناول تصميم نظام ذكي للتعرف على الوجوه باستخدام وحدة ESP32 CAM، حيث يقوم بالتقاط صورة الوجه وتحليلها ثم عرض اسم الشخص على شاشة بطريقة تلقائية. تم الاعتماد على كاميرا مدمجة ضمن وحدة ESP32 التي تتميز بقدرتها العالية على المعالجة ودعمها للاتصال اللاسلكي، مما يسمح بدمجها بسهولة في تطبيقات الذكاء الاصطناعي.

تمت برمجة النظام باستخدام بيئة Arduino IDE، مع ربطه بشاشة عرض لعرض النتائج فوراً بعد التعرف، مما يجعله مناسباً للاستعمال في أنظمة الأمن والمراقبة الذكية.

**كلمات مفتاحية:** صورة رقمية، اوردوينو، لوحة ESP32، محاكاة، برمجة.

## Résumé

Ce projet porte sur la conception d'un système intelligent de reconnaissance faciale utilisant le module ESP32-CAM. Le système capture une image du visage, l'analyse, puis affiche automatiquement le nom de la personne sur un écran. Le projet repose sur la caméra intégrée au module ESP32, connue pour sa grande capacité de traitement et sa prise en charge de la connectivité sans fil, ce qui permet une intégration facile dans des applications d'intelligence artificielle.

Le système a été programmé en utilisant l'environnement Arduino IDE, avec un écran connecté pour afficher instantanément les résultats après identification. Cette solution est particulièrement adaptée aux systèmes de sécurité et de surveillance intelligents.

**Mots-clés:** Image numérique, Arduino, Carte ESP32, Simulation, Programmation.

## Abstract

This project focuses on the design of an intelligent facial recognition system using the ESP32-CAM module. The system captures a facial image, analyzes it, and then automatically displays the person's name on a screen. It is based on the ESP32's built-in camera, which features high processing capability and wireless connectivity support, making it easy to integrate into artificial intelligence applications.

The system was programmed using the Arduino IDE environment, with a display screen connected to show the results immediately after recognition. This makes it well-suited for use in smart security and surveillance systems.

**Keywords:** Digital Image, Arduino, Board ESP32, Simulation, Programming.

## الفهرس

1	مقدمة عامة
	الفصل الأول: عموميات عن الصورة الرقمية
2	1.1 مقدمة
2	2.1 نبذة تاريخية عن معالجة الصور الرقمية
3	3.1 تعريف الصورة
3	4.1 أنواع الصورة
3	1.4.1 الصورة الثنائية
4	2.4.1 الصور ذات التدرج الرمادي
5	3.4.1 الصور الملونة
5	4.4.1 الصور متعددة الأطياف
6	5.1 معالجة الصور
6	6.1 البكسل
7	7.1 العمق اللوني
7	8.1 أنواع معالجة الصور
7	9.1 مراحل معالجة الصور الرقمية
9	10.1 تطبيقات المعالجة الرقمية للصور
11	11.1 فوائد معالجة الصور
11	12.1 خاتمة
	الفصل الثاني: دراسة نظرية لتجهيزات المشروع
12	1.2 مقدمة
122	2.2 الأردوينو
12	1.2.2 تعريف
12	2.2.2 لماذا الأردوينو

13	3.2.2	بنية لوحة الاوردوينو.....
13	4.2.2	أشهر لوحات الاردوينو.....
13	1.4.2.2	اردوينو اونو Arduino UNO.....
14	2.4.2.2	اردوينو ميغا Arduino MEGA ، Arduino MEGA 2560 ، Arduino MEGA ADK ...
15	3.4.2.2	لوحة Arduino Due.....
15	4.4.2.2	لوحة الاردوينو ليوناردو.....
16	5.4.2.2	لوحة الاردوينو ميكرو.....
16	6.5.2.2	لوحة اوردوينو ليلي باد Lily pad Arduino.....
17	5.2.2	أهم استخدامات الأردوينو.....
17	6.2.2	برنامج Arduino IDE.....
18	7.2,2	رفع البرنامج من الكمبيوتر إلى المتحكم.....
19	3.2	وحدة التحكم الدقيقة ESP32.....
19	1.3.2	تعريف.....
20	2.3.2	منافذ لوحة ESP32.....
21	3.3.2	أنواع لوحة ESP32.....
21	1.3.3.2	لوحة تطوير ESP32-DevKit-V1.....
21	2.3.3.2	لوحة تطوير YB-ESP32-S3-AMP.....
22	3.3.3.2	لوحة تطوير YB-ESP32-S3-ETH.....
22	4.3.3.2	لوحات كاميرا ESP32-CAM.....
24	5.3.3.2	لوحة تطوير YD-ESP32-S3-N8R2.....
25	6.3.3.2	لوحة تطوير T-QT مع شريحة ESP32-S3.....
25	4.3.2	كيفية برمجة لوحة ESP32.....
26	5.3.2	أيهما أفضل ؟ الاردوينو أم لوحة ESP32.....
27	4.2	الشاشات البلورية السائلية (عارض LCD).....
27	1.4.2	تعريف.....

28	.....البوريات السائلة 2.4.2
28	..... LCD المكونات الأساسية لشاشة LCD 3.4.2
28	.....LCD كيف تعمل شاشات LCD 4.4.2
29	..... OLED لحة عن تكنولوجيا OLED 4.2
30	..... خاتمة 5.2
الفصل الثالث: دراسة تطبيقية للمشروع	
31	..... مقدمة 1.3
31	..... المخطط الصندوقي 2.3
32	..... وصف النظام 3.3
32	..... العناصر الالكترونية المستعملة 4.3
34	..... تحضير قاعدة البيانات 5.3
42	..... محاكاة المشروع 6.3
43	..... شرح الدارة 7.3
44	..... صور حقيقية للمشروع 8.3
44	..... خاتمة 9.3
45	..... خاتمة عامة
46	..... المراجع

## فهرس الأشكال

رقم الصفحة	عنوان الشكل	رقم الشكل
4	صورة ثنائية	1.1
4	صورة ذات تدرج رمادي	2.1
5	صورة ملونة	3.1
6	صورة الأشعة السينية	4.1
14	لوحة أوردوينو أونو	1.2
15	لوحة اوردوينو ميغا	2.2
15	لوحة اوردوينو ديو	3.2
16	لوحة اوردوينو ليوناردو	4.2
16	لوحة اوردوينو ميكرو	5.2
17	لوحة اوردوينو ليلي باد	6.2
18	واجهة بيئة التطوير IDE	7.2
19	لوحة ESP32	8.2
23	ESP32-CAM	9.2
23	ESP32-CAM-MB	10.2
24	ESP32-CAM-USB	11.2
27	شاشة LCD	12.2
30	شاشة "0.91 OLED	13.2
31	المخطط الصندوقي لانجاز نظام التعرف على الوجه بواسطة ESP32-CAM	1.3
33	بطاقة ESP32-CAM	2.3
33	شاشة "0,91 OLED	3.3
34	محول من نوع CH340G	4.3
35	تسجيل الدخول	5.3
35	إنشاء مشروع جديد	6.3

36	جمع البيانات	7.3
37	تسمية البيانات	8.3
38	إنشاء نبضة	9.3
39	توليد المزايا	10.3
40	تدريب النموذج	11.3
41	رفع النموذج	12.3
42	محاكاة المشروع باستعمال برنامج fritzing	13.3
44	صور حقيقية للمشروع	14.3

## فهرس الجداول

رقم الصفحة	إسم الجدول	رقم الجدول
19	مواصفات اللوحة ESP32	1.2
26	الفرق بين الأردوينو وESP32	2.2
43	توصيل لوحة ESP32-CAM مع المحول	1.3
43	توصيل لوحة ESP32-CAM مع شاشة OLED	2.3

## قائمة الإختصارات

الإختصار	الإسم
IDE	Arduino Integrated Development Environment
USB	Universal Serial Bus
UART	Universal Asynchronous Receiver Transmitter
LCD	Liquid Crystal Display
OLED	Organic Light-Emitting Diode
GND	Ground
VCC	Voltage Common Collector
Wi-Fi	Wireless Fidelity

---

# مقدمة عامة

---

## مقدمة عامة

في عصر شهد تطورا رقيقا رهيبا، خرجت الأنظمة الذكية من كونها وسائل تسهل أداء المهام فقط بل أصبحت جسرا يربط الإنسان بآفاق جديدة من التطور والابتكار وهذا ما فرض علينا وأحوجنا إلى البحث العلمي والدراسة المتعمقة لمواكبة مختلف التحديات، وتعتبر البرمجة الرقمية إحدى الدعائم الرئيسية للتكنولوجيا الحديثة حيث تلعب دورا محوريا في تطويرها فيمكن للأنظمة الالكترونية القيام بمهامها من خلال كتابة تعليمات برمجية بلغة رقمية كما توفر القدرة على إنشاء تقنيات مبتكرة كالذكاء الاصطناعي وانترنت الأشياء وغيرها يعني باختصار تساهم في تطوير البرمجيات وتحسين كفاءتها بطرق مبتكرة.

ومن أبرز ما توصلت إليه الثورة الإلكترونية تقنية التعرف على الصور أو الوجوه بدقة عالية وبخوارزميات تسمح بمعالجة الصور واستخراج المعلومات منها مما يجعلها جزءا رئيسيا في العديد من التطبيقات الحياتية كأنظمة الأمن والذكاء الاصطناعي وتطبيقات الهواتف الذكية وغيرها.

ونظرا لحاجتنا إلى تطوير هذا المجال وأهميته اليوم سنقدم موضوعا بعنوان "دراسة وانجاز بطاقة إلكترونية للتعرف على الصور باستعمال ESP32-CAM"، فاخترنا وحدة التحكم ESP32 حيث مكنت من التعرف على الصور بشكل متطور وآمن وتميزت بقدرة معالجة عالية بواسطة الكاميرا المدججة بها، ولتسهيل البرمجة وتوسيع إمكانيات التحكم في لوحة ESP32 تم إضافة لوحة اردوينو ضمن بيئة تطوير مألوفة وسهلة مما ساعد في تسريع عملية البرمجة دون تعقيد، ولانجاز هذا المشروع بطريقة مرنة ارتأينا تقسيمه إلى ثلاث فصول رئيسية:

**الفصل الأول:** قدمنا فيه دراسة عامة عن الصور الرقمية وتاريخها وبعض المفاهيم الأساسية في هذا المجال كما تحدثنا عن معالجة الصور الرقمية وبعض التفاصيل المهمة فيها كمراحلها وبعض التطبيقات التي تحتاج إليها.

**الفصل الثاني:** استعراض لأهم مكونات النظام: "اردوينو" بأنواعه وبنيته وبيئة التطوير Arduino IDE، "لوحة ESP32" ودراستها دراسة شاملة إضافة إلى شاشة العرض المستعملة.

**الفصل الثالث:** في هذا الفصل قمنا بالدراسة التطبيقية للنظام مسبوقة بمحاكاة عبر برنامج fritzing.

---

الفصل الأول  
عموميات حول الصورة الرقمية

---

## 1.1 مقدمة

لم تعد الصورة الرقمية مجرد وسيلة توثيق، بل وخاصة في عصرنا هذا أصبحت جزء رئيسي في كل المجالات سواء الطب الأمن الإعلام وغيرها، لذا سلطنا الضوء على العلم الذي يهتم بإيصال الصور الرقمية إلى أحسن حال ممكن ويمكن من استغلالها بشكل صحيح واستخراج المعلومات منها، فتحدثنا في هذا الفصل عن الصور الرقمية ومعالجتها.

## 2.1 نبذة تاريخية عن معالجة الصور الرقمية

تعتمد الثورة التكنولوجية الحديثة بشكل أساسي على قدرات العقل البشري في التعامل مع الحواسيب، وشبكات الاتصال الإلكترونية، وتطوير آليات تنظيم المعلومات وتخزينها واسترجاعها، بما يحقق أقصى استفادة منها. وعند الحديث عن تاريخ الصورة الرقمية، غالباً ما يُربط تطورها بتاريخ أجهزة الحاسوب، إلا أن التطبيقات الأولى لمعالجة الصور الرقمية تعود إلى أواخر عشرينيات القرن العشرين، وتحديداً في مجال التصوير الصحفي.

في ستينيات القرن الماضي، وتحديداً عام 1962، بدأت ملامح التطور الفعلي للصورة الرقمية مع ظهور الصورة النقطية وتكنولوجيا النوافذ، إلى جانب إدخال وسيلة التفاعل الجديدة مع الحاسوب: فأرة التحكم وبحلول عام 1965، أُقيم أول معرضين لفن التصوير بالحاسوب، أحدهما في ألمانيا والآخر في الولايات المتحدة الأمريكية. وقد شارك في هذين المعرضين علماء من أمثال مايكل نول، وبيل جوليز من أمريكا، وجورج نيس، وفريدريك ناك من ألمانيا. ويُعتبر هؤلاء العلماء من الرواد الأوائل لفن التصوير الرقمي، حيث كانت أعمالهم بمثابة الشرارة التي أطلقت فن الرسوم الحاسوبية، والذي يهتم بنقل البيانات والمعلومات البصرية من وإلى الحاسوب، عبر تحويل البيانات التماثلية إلى رقمية والعكس. ومع مرور الوقت، أصبح الحاسوب أداة تعبير فني لا تقل أهمية عن أدوات الرسم التقليدية مثل الألوان الزيتية أو الفحم، حيث أتاح إمكانيات جديدة لخلق تجارب بصرية وجمالية فريدة، أحدثت ثورة في كيفية إدراكنا للفن وإنتاجه.

وفي تسعينيات القرن العشرين، وتحديداً سنة 1992، ظهر أول صالون دولي لفن التصوير الرقمي في الولايات المتحدة، ليكون ملتقى للفنانين الرقميين من مختلف أنحاء العالم. وفي عام 1994، تم تأسيس "متحف فنون الكمبيوتر" (Museum of Computer Art - MOCA)، والذي يُعد من أكبر

متاحف الفن الرقمي على الإنترنت، مما أتاح لمحبي هذا الفن فرصة الاطلاع على معروضاته من أي مكان في العالم، وفي أي وقت، متجاوزاً بذلك حدود الزمان والمكان.

من جهة أخرى، ساهم علماء الرياضيات في تطور الفن الرقمي من خلال تحويل المعادلات إلى أشكال بصرية. ومن أبرز هؤلاء، العالم الرياضي ميشيل بارنسلي الذي قام بترجمة معادلات رياضية إلى أرقام مركبة، نتج عنها تكوينات زخرفية مبهرة. كما استخدمت معادلات العالم الإنجليزي إسحق نيوتن في تصميم تركيبات فنية رقمية متميزة. وهكذا، تحولت الأرقام والمعادلات إلى عناصر زخرفية وجداريات فنية من خلال ما يُعرف بالهندسة الفراكتالية، وهي تقنية فنية تقوم على تكرار التفاصيل الصغيرة داخل الشكل الأكبر، مما يمنحه بنية جمالية معقدة ومتناسقة في آنٍ واحد [1].

### 3.1 تعريف الصورة

يمكن تعريف الصورة على أنها دالة ثنائية الأبعاد  $f(x,y)$ ، حيث  $x$  و  $y$  هما إحداثيات الفضاء (السطح)، وتُعرف السعة (أو القيمة) للدالة  $f$  عند أي زوج من الإحداثيات  $(x, y)$  بأنها شدة الصورة أو مستوى اللون الرمادي عند تلك النقطة. عندما تكون قيم  $x$  و  $y$  وقيم الشدة  $f$  كلها كميات محدودة ومنفصلة، نطلق على الصورة اسم صورة رقمية [2].

تعتبر الصورة الرقمية بمثابة تمثيل مفصل للبيانات التي تمتلك معلومات مكانية (التخطيط) ومعلومات كثافة (اللون) [3].

### 4.1 أنواع الصورة

#### 1.4.1 الصورة الثنائية

تستخدم الصور الثنائية بت (bit) واحد فقط لكل موقع في مصفوفة الصورة الرقمية إما 1 أو 0 والـ (bit) هو أصغر من البايت (Byte) والبايت هو وحدة معلومات رقمية في الحاسوب وعادة ما يتكون كل بايت من 8 بت، وكل بكسل في الصورة الثنائية يجب أن يكون أحد اللونين إما أبيض أو أسود، ولذلك تكون قدرتها محدودة في التعامل مع الصور الفوتوغرافية لعدم قدرتها على تمثيل الظلال المتوسطة للصور الرمادية.



الشكل (1.1): صورة ثنائية

## 2.4.1 الصور ذات التدرج الرمادي

يمكن تمثيل الصور ذات التدرج الرمادي كصور ثنائية في فضاء ثلاثي الأبعاد، حيث يمثل البعد الثالث السطوع. يمكن بعد ذلك عرض الصورة ذات التدرج الرمادي كسطح ثلاثي الأبعاد، مع ارتفاع في كل نقطة مساو لقيمة السطوع. تتكون الصور ذات اللونين الأبيض والأسود وحدات البكسل، كل منها يحمل رقماً واحداً يتوافق مع المستوى الرمادي للصورة في موقع معين، تمتد هذه المستويات الرمادية على نطاق كامل في سلسلة من التدرج اللوني من الأسود إلى الأبيض، باقتراض 256 مستوى رمادي كالاتي (0-255)، يمكن تخزين كل بكسل بالأبيض والأسود في بايت واحد (8 بت) من الذاكرة.



الشكل (2.1): صورة ذات تدرج رمادي

### 3.4.1 الصور الملونة

تتكون الصورة الملونة من وحدات بكسل، يحتوي كل منها على ثلاثة أرقام مطابقة لمستويات الأحمر والأخضر والأزرق للصورة في موقع معين، بافتراض 256 مستوى، يمكن تخزين كل بكسل لون في ثلاثة بايت (24 بت) من الذاكرة. الحظ أنه بالنسبة للصور من نفس الحجم، فإن النسخة بالأبيض والأسود ستستخدم ذاكرة أقل بثلاث مرات من النسخة الملونة.



الشكل (3.1): صورة ملونة

### 4.4.1 الصور متعددة الأطياف

بعض أنواع الصور متعددة الأطياف هي:

- ✓ صور الأشعة تحت الحمراء.
- ✓ تشخيص الصور الطبية.
- ✓ صور الأشعة السينية [4].



الشكل (4.1): صورة الأشعة السينية

### 5.1 معالجة الصور

معالجة الصورة (Image Processing) هي تمثيل للصور الثنائية الأبعاد على الحاسوب بواسطة الصفر والواحد (01)، وتتكون كل صورة رقمية على الحاسوب من البكسل (Pixel) وهو أصغر وحدة في الصورة وكل صورة تحتوي على صفوف و أعمدة من البكسلات وكلها زادت عدد البكسلات كلما كانت الصورة أوضح.

تعرف معالجة الصورة أيضا بأنها أحد فروع علم الحاسوب (المعلوماتية)، تهتم بإجراء عمليات على الصور بهدف تحسينها طبقاً لمعايير محددة أو استخلاص بعض المعلومات منها [5].

### 6.1 البكسل

هو أصغر وحدة متميزة في الصورة التي تمتلك إضاءة ولونا محددين. تتحدد أبعاده بناءً على عدد الخطوط التي تشكل نمط المسح (الشبكة النقطية التي تُكوّن الصورة) والدقة على كل خط. في أكثر أشكال الرسوميات الكمبيوتر شيوعاً، يتم عرض الآلاف من البكسلات الصغيرة التي تشكل الصورة الفردية على شاشة العرض كنقاط مضيئة، وعند النظر إليها من مسافة، تظهر كصورة مستمرة، كما يمكن أن يكون البكسل أصغر عنصر في جهاز حساس للضوء، مثل الكاميرات التي تستخدم الأجهزة المقترنة بالشحن [6].

### 7.1 العمق اللوني

هو مصطلح في الرسوميات الحاسوبية يصف عدد البتات التي تستخدم في تمثيل لون بكسل واحد في صورة رقمية. يطلق على هذا المبدأ أيضاً اسم بت لكل بيكسل (بالإنجليزية bits per pixel) [7].

### 8.1 أنواع معالجة الصور

هناك خمسة أنواع رئيسية لمعالجة الصور:

- ✓ التصور: إيجاد الأشياء التي لا تظهر في الصورة.
- ✓ التعرف: التمييز أو الكشف عن الأشياء في الصورة.
- ✓ التحسين والاستعادة: إنشاء صورة محسنة من الصورة الأصلية.
- ✓ التعرف على الأنماط: قياس الأنماط المختلفة حول الأشياء في الصورة.
- ✓ استرجاع الصور: تصفح والبحث عن الصور من قاعدة بيانات كبيرة للصور الرقمية التي تشبه الصورة الأصلية [8].

### 9.1 مراحل معالجة الصور الرقمية

المرحلة الأولى: الحصول على الصورة

إن الخطوة الأولى من الخطوات سير عمل معالجة للصورة هي تحميل الصورة أولاً لمعالجتها. يمكن تحميل أي صورة من أي مصدر نشاء ولكن الأهم من ذلك أن الصورة التي علينا تحميلها يجب أن تكون غير معالجة رقياً لأن النتيجة حينئذ لن تكون مختلفة عن ذي قبل.

المرحلة الثانية: تحسين الصورة

يوجد العديد من الأشياء التي يمكن فعلها من أجل تحسين الصورة، من هذه الأشياء، شخذ الصورة وضبط السطوع والتباين فيها. كل تلك التي ذكرناها سابقاً تؤدي إلى تحسين الصورة وستجعل منها مختلفة وأفضل من ذي قبل وسيلاحظ ذلك بشكل كبير نسبياً.

#### المرحلة الثالثة: استعادة الصورة

هي عملية تصحيح الصورة الأصلية إذا كانت فيها عيوب أو تشويش بسبب طريقة الالتقاط، النقل أو التخزين، مثلاً صورة فيها ضباب أو تشويش (يتم تجاهل هذه العملية رغم أهميتها لأن عملية التحسين تكون قد غيرت تغيراً جذرياً في الصورة).

#### المرحلة الرابعة: معالجة الصور الملونة

في هذه المرحلة يتم معالجة الصورة من ناحيتين إما كصورة مفهرسة أو كصورة من نوع RGB أما إذا أردنا المعالجة بالطريقة المتعددة الحلقات فنحن أمام طريق الموجات. والموجات هذه هي موجات صغيرة ذات طول محدد، الهدف في تحويل الموجة هو الحصول على ما يسمى بالتردد الزمني وطريقة الموجات هذه تؤيد إلى معالجة الصور بدقات مختلفة ومرتفعة.

#### المرحلة الخامسة: مرحلة الضغط

إن مرحلة الضغط هذه ضرورية جداً بعد عمل كل تلك المراحل السابقة. ذلك من أجل تقليل مساحة التخزين المطلوبة لحفظ صورة أو عرض النطاق الترددي المطلوب لنقل الصورة. هذه العملية مفيدة جداً وخاصة عند عرضها على الأنترنت فإذا كان حجمها كبيراً فإن الضغط يسهل ظهورها عبر الأنترنت بحجم صغير وبدقة عالية كما يطلبه الجمهور.

#### المرحلة السادسة: المعالجة الصرفية والتجزئة

تقوم هذه المرحلة باستخراج مكونات الصورة المميزة منها من أجل تمثيل ووصف شكل الصورة. ومن أجل القيام بهذه المرحلة يوجد عدة عمليات يمكن القيام بها مثل التعرية والتجدد. وهي ما تسمى أيضاً بمخرجات المعالجة المورفولوجية والتي تعني سمات الصورة.

أما عملية التجزئة فهي تعني تقسيم الصورة الرقمية إلى عدة أجزاء، الهدف منها تحديد موقع كائنات الصورة وصدورها.

المرحلة السابعة: مرحلة التمثيل والوصف

تقوم هذه المرحلة أساساً على تحويل البيانات إلى نموذج مناسب من أجل معالجتها على الكمبيوتر. تتم عملية التمثيل من خلال عدة طرق أولها، التمثيل الحدودي الذي يستخدم عندما يكون التركيز على الخصائص المميزة للشكل الخارجي مثل زوايا الصورة. ثانياً، التمثيل الإقليمي الذي يستخدم عندما يكون التركيز على الخصائص الداخلية مثل الملمس للصورة. أما الوصف فهو يتعامل أساساً مع استخراج الصفات التي تظهر المعلومات ذات الأهمية. وهو يستخدم لتمييز فئة معينة من الكائنات عن غيره [9].

10.1 تطبيقات المعالجة الرقمية للصور

تُستخدم معالجة الصور في مجالات متنوعة تَمَسُّ العديد من الأنشطة البشرية، من الصناعة إلى الطب والدفاع وغيرها. فيما يلي أهم التطبيقات:

- أنظمة الفحص البصري الآلي: تُستخدم لتحسين جودة المنتجات في الصناعة:
  - تفتيش المصابيح: كشف عدم انتظام في أسلاك الفتيلة باستخدام صور ثنائية.
  - كشف المكونات المعطلة: باستخدام صور الأشعة تحت الحمراء لتحديد الأجزاء التي تصدر حرارة غير طبيعية.
  - فحص الأسطح: لاكتشاف العيوب في الأسطح المعدنية مثل الصفائح المدرفلة، عبر تقنيات مثل كشف الحواف وتحليل الملمس.
- تفسير المشاهد بالاستشعار عن بُعد:
  - تحليل صور الأرض الملتقطة بواسطة الأقمار الصناعية أو الطائرات
  - يُستخدم لاستخراج معلومات عن الموارد الطبيعية: زراعية، مائية، معدنية، غابية،... إلخ.
  - يُطبَّق في مجالات مثل تخطيط المدن، مراقبة الفيضانات، وتبعية الإنتاج الزراعي.

- التصوير الطبي الحيوي: تُستخدم صور الأشعة لتشخيص الأمراض مثل:
  - أمراض الرئة: فحص الأشعة السينية لتحديد الشذوذ في الرئتين.
  - أمراض القلب: قياس حجم وشكل القلب لتحليل المشاكل.
  - الماموغرام الرقمي: تحليل الأورام في الثدي باستخدام تقنيات مثل تعزيز التباين واستخراج السمات.
- المراقبة الدفاعية (Defense Surveillance): مراقبة الأرض والمحيطات باستخدام الصور الجوية:
  - استخراج وتحديد الأجسام مثل السفن.
- الاسترجاع المعتمد على المحتوى: استرجاع الصور من قواعد بيانات ضخمة اعتماداً على محتواها:
  - تُستخدم خصائص مثل: الشكل، اللون، النسيج، لترميز الصور.
  - يُساعد في إنشاء أنظمة بحث فعالة عن الصور الرقمية.
- تتبع الأجسام المتحركة: تتبع الأجسام في الزمن الحقيقي لأغراض تحليل الحركة:
  - تعتمد على طريقتين: التتبع بالعرف، والتتبع بالحركة.
  - تُستخدم فلاتر تنبؤية مثل: كالمان، كالمان الموسع، وفلاتر الجسيمات.
- ضغط الصور والفيديو: تقليل حجم بيانات الصور والفيديو لتسهيل التخزين والإرسال:
  - يتم التخلص من التكرار البصري غير الضروري في الصور.
  - يقلل من متطلبات التخزين وعرض النطاق الترددي، وهو ضروري لتطبيقات مثل التلفاز عالي الدقة والطب الرقمي [10].

### 11.1 فوائد معالجة الصور

لتطبيق تقنيات معالجة الصور تأثير هائل على العديد من المؤسسات التقنية. وفيما يلي بعض فوائد معالجة الصور:

- ✓ يمكن إتاحة الصورة الرقمية بأي تنسيق مطلوب (صورة محسنة، صورة سلبية... إلخ).
- ✓ يساعد على تحسين الصور للتفسير البشري.
- ✓ يمكن معالجة المعلومات واستخراجها من الصور لتفسير الآلة.
- ✓ يمكن معالجة وحدات البيكسل الموجودة في الصورة إلى أي كثافة وتباين مرغوب فيهما.
- ✓ يمكن تخزين الصور واسترجاعها بسهولة [8].

### 12.1 خاتمة

تعرفنا في هذا الفصل على الصور الرقمية ومعالجتها، ونظرا لأهميته في كل المجالات يمكننا القول أنه سيشهد تطورا كبيرا في المستقبل مما يعزز من قدرته على التعامل مع تحديات جديدة ومبتكرة.

---

الفصل الثاني  
دراسة نظرية لتجهيزات المشروع

---

## 1.2 مقدمة

إن من أهم ما توصلت إليه الهندسة الإلكترونية في هذا العصر دمج البرمجيات مع العناصر الكهربائية والإلكترونية، حيث أتاح لنا هذا التطور التحكم في تركيب معين أو تغيير مبدأ عمله بصفة مرنة وسريعة وذلك عن طريق لوحات البرمجة (لوحات التطوير) مثل الاردوينو الذي انتشر انتشارا واسعا وأيضا ESP32، وسنتطرق في هذا الفصل إلى دراسة مكوناتها ومختلف أنواعها، كما تطرقنا إلى الشاشات البلورية السائلة أو ما يسمى بالعارض LCD و شاشة OLED.

## 2.2 الأردوينو

## 1.2.2 تعريف

له عدة تعاريف تتفق في المعنى نذكر منها هذين التعريفين:

لوحة الاردوينو هي لوحة الكترونية تم تصميمها لإنشاء الدارات والمشاريع الالكترونية المختلفة بكل سهولة تسم هذه اللوحة بكونها قابلة للبرمجة (programmable board)، وهذا يعني انه يتم برمجتها من قبل المستخدم لتؤدي الغرض المطلوب منها حيث تتم البرمجة بلغة Arduino C وهي لغة سهلة ومتوفرة بشكل مجاني ومفتوحة المصدر [10].

لوحة إلكترونية تستخدم لتطوير الكثير من الأفكار والمشاريع المتعلقة بالتحكم الآلي بصورة سهلة وبسيطة عن طريق استخدام لغة برمجة مفتوحة المصدر Arduino C حيث يتم برمجة المتحكم الموجود في اللوحة باستخدام برنامج خاص [11].

## 2.2.2 لماذا الأردوينو

في الحقيقة يوجد العديد من المتحكمات الالكترونية التي تتوفر على إمكانيات قوية، سنذكر الآن بعض ما يميز الاردوينو عنها:

✓ البساطة: تناسب احتياجات الجميع، محترفين، هواة وطلبة وغيرهم.

✓ أقل سعرا مقارنة مع بقية اللوحات من نفس النوع.

✓ متعدد المنصات حيث له القدرة العمل على windows، Linux، Moc OS بينما أغلب المتحكمات الأخرى تعمل فقط على ويندوز.

✓ مكتوب بلغة C++ و متاح تحميله للجميع وبإمكان المبرمجين التعديل عليه وفق احتياجاتهم [12].

### 3.2.2 بنية لوحة الاردوينو

تنقسم بنيته إلى أربعة أقسام عامة :

(1) قسم المتحكم: والذي يضم المتحكم وكل ما يلزم لعمل المتحكم بالشكل الصحيح (مذبذب خارجي مع المكثفات الخاصة بها، زر إعادة الضبط reset، مكثفات الاستقرار الكهربائي على مداخل التغذية، أرجل أقطاب الدخل والخرج العامة).

(2) قسم الاتصال مع الحاسوب: وهي شريحة لتأمين الاتصال بين الحاسوب ولوحة الاردوينو أي بين البروتوكولين UART<>USB (غالبا تكون هذه الشريحة إما IC:FTDI أو متحكم (ATmega8

(3) قسم التغذية: لتأمين وضبط تغذية المتحكم ويتم ذلك عبر منظمات جهد لتعطي على لوحة الاردوينو جهد بقيمة 5V & 3.3V.

(4) ملحقات على لوحة الاردوينو: كالأزرار المحظية ومداخل قراءة بطاقة الذاكرة أو حساس أو أي نوع آخر وتكون هذه الملحقات لتسهيل التوصيل مع الاردوينو، وتوفر هذه الملحقات على لوحات دون غيرها [13].

### 4.2.2 أشهر لوحات الاردوينو

يأتي الاردوينو في عدة أشكال تختلف في الحجم والأداء والمنافذ، اخترنا المرور على هذه الأنواع:

#### 1.4.2.2 اردوينو اونو Arduino UNO

هو النوع الأنسب للبدء بالتعلم على الاردوينو، يوجد منه ثلاث إصدارات حاليا بينها اختلافات بسيطة، يحتوي على 14 منفذ وأهم عنصر فيه هو المتحكم الاصغري الرئيسي الذي يتم برمجته تحديدا

ثم يقوم هو الآخر بتنفيذ البرنامج المخزن فيه وهو من نوع ATmega328 حيث يتصل بمذبذب تردده 16MHz لذا يمكنه تنفيذ 16 مليون تعليمة في ثانية واحدة، لا يمكن للمتحكم الرئيسي ATmega328 تبادل البيانات مع الحاسب عبر منفذ USB لذا تم إضافة المتحكم الأصغر من نوع ATmega16U2 الذي يتوفر على وحدة اتصال أساسية USB في بنيته الداخلية [11].

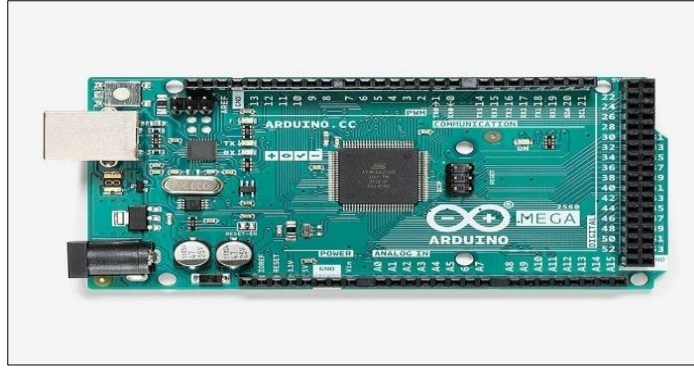
يمكن تعريفه أيضا أنه: دائرة إلكترونية صغيرة تستخدم في برمجة متحكم من نوع ATmega328P وتوفر منافذ لتوصيل المكونات الإلكترونية إلى هذه المتحكم عن طريق 14 منفذ من النوع الرقمي (مداخل/مخارج) [12].



الشكل (1.2): لوحة أوردوينو أونو

#### 2.4.2.2 اردوينو ميغا Arduino MEGA 2560، Arduino MEGA 2560، ADK Arduino MEGA

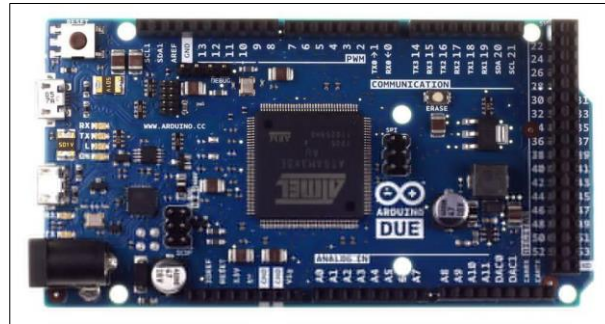
أكبر حجما من اردوينو اونو حيث تصل عدد المنافذ فيه إلى 54 منفذ، يحتوي Arduino MEGA على متحكم أصغر من نوع ATmega1280 لذا فان سعة ذاكرة البرنامج تصل إلى 128KB بينما يحتوي اللوحان الأخران على متحكم من نوع ATmega2560 بالتالي سعة الذاكرة 256KB [11].



الشكل (2.2): لوحة أوردوينو ميغا

### 3.4.2.2 لوحة Arduino Due

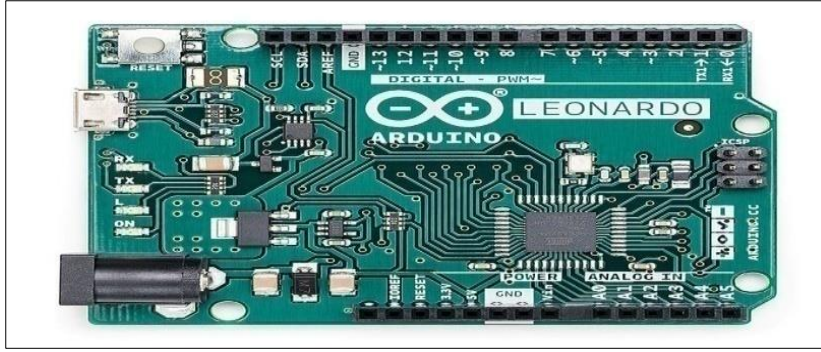
تشبه لوحة Arduino MEGA ADK من حيث الشكل وتختلف عنها في البنية الإلكترونية، تستخدم متحكم من نوع AT91SAM3X8E الذي يعتمد على بيئة RAM بـ 32 بت، مع سرعة معالج تصل إلى 84MHz، تعمل اللوحة بجهد مقداره 3.3 V.



الشكل (3.2): لوحة أوردوينو ديو

### 4.4.2.2 لوحة الأوردوينو ليوناردو

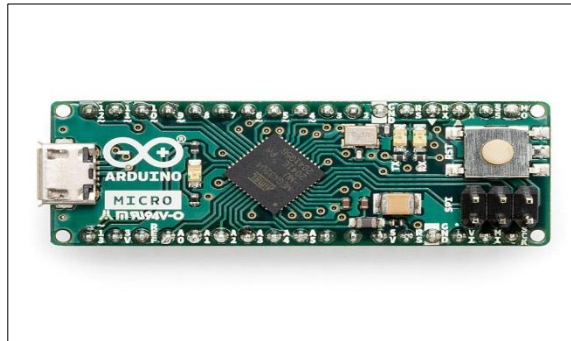
تشبه لوحة الاونو من حيث الشكل وتختلف عنها بالمتحكم الأصغري الرئيسي المستخدم وهو ATmega32u4 الذي يحتوي على وحدة اتصال USB مدمجة لذا لا يحتاج إلى متحكم ثانوي.



الشكل (1.2): لوحة اوردوينو ليوناردو

### 5.4.2.2 لوحة الاردوينو ميكرو

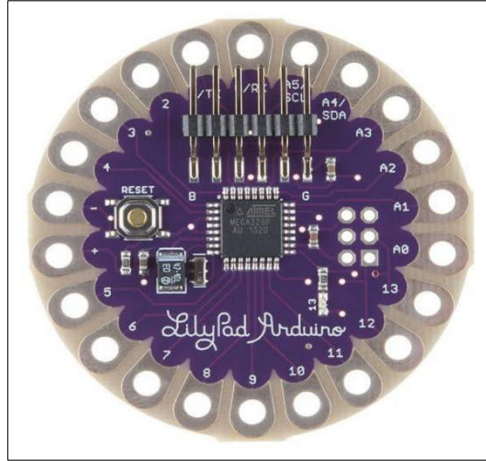
تعتبر لوحة مصغرة للوحة اوردوينو ليوناردو ولها نفس مزاياه وخواصه باستثناء أبعادها ووزنها، تم تصميمها بحيث يمكن استخدامها بسهولة مع لوحة التجارب.



الشكل (5.2): لوحة اوردوينو ميكرو

### 6.5.2.2 لوحة اوردوينو ليلي باد Lily Pad Arduino

تم تصميمها لتعمل مع المنسوجات الالكترونية، تقدم نفس وظائف لوحات اوردوينو الأخرى مع مزايا أخرى مثل خفة الوزن ودائرية الشكل ولها منافذ خرج كبيرة بهدف تسهيل عملية الاتصال والربط مع الملابس [11].



الشكل (6.2): لوحة اوردوينو ليلي باد

## 5.2.2 أهم استخدامات الأردوينو

له الكثير من الاستخدامات العملية نذكر منها:

- مشاريع التحكم في تشغيل الأجهزة .
- صنع رجل آلي.
- جهاز انذار الحرائق.
- سقي المزروعات تلقائيا .
- أجهزة تحديد موقع السيارات أو الأشخاص عبر الأقمار الصناعية.
- مشاريع قياس درجة الحرارة والرطوبة وغيرها.
- التحكم بأجهزة المنزل عبر الهاتف [14].

## 6.2.2 برنامج Arduino IDE

بيئة التطوير المتكاملة للأردوينو (IDE) Arduino integrated development enviroment:

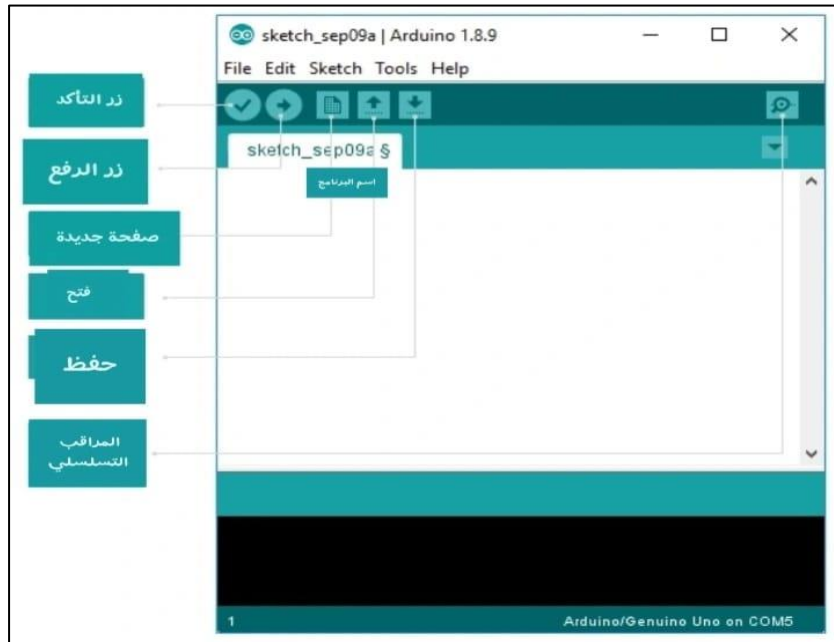
هي البرنامج الذي من خلاله يتم كتابة الكود البرمجي للوحة الاردوينو بلغة C وتحميله إليها. تسم بيئة التطوير بالبساطة وسهولة التعامل وتحتوي على كل ما يحتاجه المبرمج لكتابة الشيفرة، وتعمل على أنظمة التشغيل المختلفة Linux، Mac OSX، Windows.

يمكن تحميله من الموقع الرسمي لمطوري الاردوينو من الرابط التالي:

<https://www.arduino.cc/en/Main/Software>

بعد فتح الرابط نختار نظام التشغيل ومن ثم just download ليبدأ التحميل مباشرة ومجاناً.

قم بفك الضغط للملف المحمل الذي يحتوي على برنامج بيئة التطوير وتعريف ومكتبات وأمثلة وغيرها، قم بتشغيل برنامج Arduino.exe لتعمل بيئة التطوير مباشرة دون الحاجة إلى تنصيب [15].



الشكل (7.2): واجهة بيئة التطوير IDE

## 7.2.2 رفع البرنامج من الكمبيوتر إلى المتحكم

- نتأكد من توصيل الاردوينو بالكمبيوتر عن طريق سلك ال USB.
- نختار المنفذ المتصل باللوحة عن طريق برنامج الاردوينو من الكمبيوتر كالتالي:  
نذهب الى أدوات ← منفذ ← المنفذ الخاص بالاردوينو

choose → Tools -> port

(إذا لم يوجد المنفذ الخاص بالاردوينو تأكد من توصيل الاردوينو أو تثبيت البرنامج بشكل صحيح، في حالات كثيرة تكون المشكلة عدم تثبيت ال driver الخاص بالاردوينو).

- نختار نوع اللوحة الموجودة لدينا عن طريق:

أدوات ← اللوحة ← نختار اسم اللوحة (Arduino Uno) .

نضغط upload

- سيظهر لك في الأسفل معلومات التحميل، وإن كان هناك خطأ سيكتب لك سببه غالباً.

- في حال كان كل شيء على ما يرام، ستظهر لك عبارة done uploading [15].

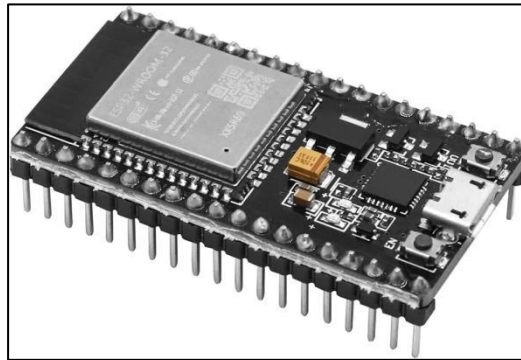
### 3.2 وحدة التحكم الدقيقة ESP32

#### 1.3.2 تعريف

ESP32 هو سلسلة وحدات تحكم مركزية ونظام على شريحة منخفضة التكلفة ومنخفضة الطاقة

مع ميزة الواي فاي ووضع بلوتوث مزدوج. تم تطويره بواسطة شركة Espressif System

الصينية [17].



الشكل (8.2): لوحة ESP32

## الجدول (1.2): مواصفات اللوحة ESP32

المواصفات	القيمة
المعالج	ثنائي النواة 32 بت من نوع Xtensa LX6
تردد المعالج	160 ميغاهرتز أو 240 ميغاهرتز
ذاكرة RAM	520 كيلوبايت
ذاكرة ROM	448 كيلوبايت
Wi-Fi	802.11 b/g/n

## 2.3.2 منافذ لوحة ESP32

تحتوي على عدد كبير من المنافذ التي يمكن استخدامها في توصيل المكونات الإلكترونية والتحكم بها، ويتم تصنيف هذه الأقطاب إلى عدة فئات:

1. منافذ التيار الكهربائي: هذه الأقطاب مخصصة لتوصيل مصدر التيار الكهربائي إلى اللوحة، وتستخدم عادة لتوصيل بطارية خارجية أو كابل USB باللوحة، تحمل هذه المنافذ العلامات Vin و VUSB و 5V و 3.3V.

2. منافذ المداخل التماثلية: تستخدم لتوصيل المستشعرات والأجهزة التي تعمل بإشارات تماثلية، وتحمل العلامات A0 و A1 و A2 وهكذا حتى A15.

3. منافذ المداخل الرقمية: تستخدم هذه الأقطاب لتوصيل المكونات الإلكترونية التي تعمل بإشارات رقمية، وتحمل العلامات D0 و D1 وهكذا حتى D39.

4. منافذ المخارج الرقمية: تستخدم لتوصيل المكونات الإلكترونية التي تعمل بإشارات رقمية، وتستخدم عادة لتشغيل المحركات والمصابيح وغيرها، وتحمل العلامات D0 و D1 وهكذا حتى D39.

5. منافذ المخارج PWM: تستخدم هذه الأقطاب لتوصيل المكونات الالكترونية التي تحتاج إشارات PWM وهي عبارة عن إشارات تتغير في شدة الإشارة مع الوقت وتحمل العلامات PWM0 و PWM1 وهكذا حتى PWM19.

6. منافذ SPI: تستخدم هذه المنافذ لتوصيل المكونات الإلكترونية التي تعمل بروتوكول SPI وتستخدم عادة في تحويل الشاشات LCD وذاكرة التخزين الفلاش وغيرها، وتحمل العلامات MISO و MOSI و SCK و CS [18].

### 3.3.2 أنواع لوحة ESP32

نذكر منها :

#### 1.3.3.2 لوحة تطوير ESP32-DevKit-V1

لوحة تطوير ESP32-DevKit-V1 هي لوحة تطوير ذات حجم صغير ومستوى دخول منخفض وهي جزء من سلسلة ESP32 DevKit من شركة Espressif وقد كانت متاحة لعدة سنوات. تحتوي اللوحة على وحدة ESP-WROOM-32 مع ذاكرة فلاش بسعة 4 ميجابايت وبدون PSRAM، وزرين BOOT و EN/RESET، ومصباحين LED واحد للطاقة والآخر لـ GPIO2، ومنظم جهد LDO مثل AMS1117-3.3، NCP1117-3.3، لتحويل الجهد VIN إلى 3.3 فولت، ومنفذ microUSB، وعدد كبير من منافذ GPIO (منافذ الإدخال/الإخراج العامة).

#### 2.3.3.2 لوحة تطوير YB-ESP32-S3-AMP

لوحة تطوير YB-ESP32-S3-AMP هي لوحة تطوير متعددة الاستخدامات تعتمد على معالج ESP32-S3 من شركة Espressif، تأتي هذه اللوحة بإصدارين مختلفين (Rev.2/Rev.3) وتتميز بوحدة ESP32-S3-WROOM-1-N8R2 (ذاكرة فلاش 8 ميجابايت، 2 ميجابايت PSRAM، هوائي PCB لوصلة WiFi) وهي متاحة على منصات البيع مثل eBay و Ricardo.ch. بالإضافة إلى شريحة ESP32-S3، توفر اللوحة فتحة لبطاقة microSD، معززين صوتيين من نوع USB-C MAX98357AI2S PCM Class D (للقنوات الصوتية اليمنى واليسرى)، موصل

لتحميل البرمجيات، إخراج تسلسلي و/أو تغذية الطاقة إلى اللوحة، ومصباحين LED لحالة التشغيل، بالإضافة إلى منافذ GPIO للاستخدام الحر.

الإصدار 2 يحتوي على شريحة جسر USB CH340 بين USB-C و ESP32-S3. أما في اللوحات التي تحمل الإصدار 3، فإن USB-C متصل مباشرة بشريحة ESP32-S3.

### 3.3.3.2 لوحة تطوير YB-ESP32-S3-ETH

لوحة تطوير YB-ESP32-S3-ETH هي لوحة تطوير متعددة الاستخدامات تعتمد على معالج ESP32-S3 من شركة Espressif، تأتي حالياً بثلاثة إصدارات مختلفة:

1. مع وحدة ESP32-S3-WROOM-1U-N8R8 (8 ميجابايت فلاش / 8 ميجابايت PSRAM).
2. مع وحدة ESP32-S3-WROOM-1U-N4 (4 ميجابايت فلاش).
3. بدون وحدة ESP32-S3-WROOM-1(U).

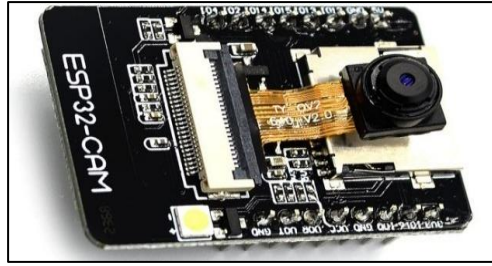
توفر جميع الإصدارات موصل Ethernet RJ45، وشريحة جسر Ethernet PHY من Wiznet W5500، ومصباحين LED لحالة التشغيل، وموصلين microUSB لتحميل البرمجيات، والإخراج التسلسلي، والتصحيح، وتغذية الطاقة للوحة. اللوحات متاحة حالياً على منصات البيع مثل eBay و Ricardo.ch.

### 4.3.3.2 لوحات كاميرا ESP32-CAM

تأتي لوحات كاميرا ESP32-CAM AI-Thinker بأنواع مختلفة. جميع اللوحات تحتوي على الأقل على وحدة ESP32-S، ومنفذ كاميرا OV2640، وشريحة PSRAM (4 ميجابايت)، و LED عالي القدرة، تأتي على ثلاث أنواع:

1. ESP32-CAM: أصغر حجماً وبدون ميزات إضافية، يجب برمجتها باستخدام وحدة

USB-TTL منفصلة متصلة بها.



الشكل (9.2): ESP32-CAM

2. ESP32-CAM-MB: تأتي مع لوحة شطيرة (قابلة للإزالة) تحتوي على شريحة جسر USB-TTL (CH340C)، وأزرار IO0 و RST، ومنفذ microUSB. تُستخدم اللوحة الإضافية لبرمجة لوحة الكاميرا وطباعتها الإخراج التسلسلي.



الشكل (9.2): ESP32-CAM-MB

3. ESP32-CAM-USB: لوحة واحدة، أكبر قليلاً من لوحة ESP32-CAM، تحتوي على شريحة جسر USB-TTL (CH340C)، وأزرار FLASH و RST، ومنفذ microUSB، بما أن هذه اللوحة لا تحتوي على منطق إعادة الضبط، يجب استخدام الأزرار "Flash" و "Reset" لدخولها في وضع التحميل وإعادة ضبطها بعد التحميل الناجح.



الشكل (11.2): ESP32-CAM-USB

### 5.3.3.2 لوحة تطوير YD-ESP32-S3-N8R2

لوحة YD-ESP32-S3-N8R2 هي نسخة مقلدة من لوحة ESP32-S3-DevKitC-1، وتتميز بمجموعة من المواصفات المتقدمة التي تجعلها مناسبة لمجموعة متنوعة من المشاريع.

المميزات الرئيسية:

- شريحة ESP32-S3: تحتوي على شريحة ESP32-S3 مع دعم Wi-Fi و Bluetooth 5 و .LE
- منافذ USB-C: تضم اللوحة منفذين USB-C مما يوفر خيارات مرنة للاتصال وتغذية الطاقة.
- ذاكرة فلاش 8 MB و 2MB PSRAM: توفر الذاكرة اللازمة لتشغيل التطبيقات المتطورة.
- منافذ GPIO وفيرة: توفر اللوحة العديد من منافذ GPIO للإستخدام في تطبيقات مختلفة.
- إضاءة RGB WS2812B: تحتوي على LED RGB من نوع WS2812B.

الإصدارات المختلفة:

- N16R8: مع 16MB Flash و 8MB PSRAM.
- N8R8: مع 8MB Flash و 8MB PSRAM.
- بالإضافة إلى إصدارات أخرى متنوعة.

تعتبر هذه اللوحة مثالية لمشاريع الإنترنت (IOT) التي تتطلب الاتصال اللاسلكي ومعالجة قوية.

### 6.3.3.2 لوحة تطوير T-QT مع شريحة ESP32-S3

تعد لوحة T-QT لوحة تطوير رئيسية تحتوي على شريحة ESP32-S3، وهي مجهزة بشاشة LCD ملونة بحجم 0.85 بوصة مع دقة عرض عالية، بالإضافة إلى زرین قابلين للبرمجة. تتميز بتصميم تخطيط مشابه لـ T-Display، مما يسهل استخدامها في مشاريع مماثلة.

اللوحة تدعم الاتصال والتعديل المباشر عبر USB باستخدام ESP32-S3، مما يجعلها سهلة الاستخدام من حيث البرمجة وتبادل البيانات [19].

### 4.3.2 كيفية برمجة لوحة ESP32

يمكن برمجة لوحة ESP32 باستخدام العديد من الأدوات المنتجة، ومن أشهر هذه الأدوات:

1. Arduino IDE: ويتطلب ذلك تثبيت المكتبات المناسبة وإعداد المنفذ الذي يتم توصيل اللوحة به.

2. MicroPython: يمكن استخدام هذه اللغة لبرمجة لوحة ESP32، وتعتبر هذه اللغة أسهل في التعلم والاستخدام مقارنة بلغة الوردوينو.

3. Visual Studio Code: برنامج يمكن استخدامه لبرمجة ESP32 باستخدام لغات البرمجة المختلفة ويتوفر فيه العديد من الملحقات التي تسهل عملية البرمجة.

4. PlatformIO: يمكن استخدام هذه البيئة المتاحة كإضافة لبرنامج Visual Studio Code لبرمجة لوحة ESP32، وتوفر هذه البيئة العديد من المميزات مثل إمكانية إضافة المكتبات بسهولة وتشغيل الأكواد على أنظمة تشغيل مختلفة.

يتم برمجة لوحة ESP32 باستخدام لغات البرمجة المختلفة مثل لغة C و C++ و Python، ويتم تحميل البرنامج المنشأ على اللوحة باستخدام برامج لتحميل البرامج المتاحة مثل برنامج Csptool أو برنامج Flash Downlaodtools المقدم من شركة Espressif التي تنتج هذه اللوحة [18].

### 5.3.2 أيهما أفضل ؟ الأردوينو أم لوحة ESP32

الجدول (1.2): الفرق بين الأردوينو وESP32

ESP32	أوردوينو
تم تصميم ESP32 مع العديد من أوضاع الطاقة لتحسين استهلاك الطاقة، بما في ذلك وضع السكون العميق الذي يستهلك فقط بضعة ميكرو أمبير فقط، وهذا يجعله مثاليًا لتطبيقات إنترنت الأشياء التي تعمل بالبطارية حيث يكون الحفاظ على الطاقة أمرًا بالغ الأهمية	معظم لوحات Arduino لا تحتوي على ميزات إدارة الطاقة المتقدمة. بينما توجد بعض لوحات Arduino منخفضة الطاقة، فإن استهلاك الطاقة النموذجي يكون عادةً أعلى مقارنة بـ ESP32.
يوفر اتصال Wi-Fi و Bluetooth مدمجين (بما في ذلك BLE)، مما يجعله مناسبًا للغاية للتواصل اللاسلكي ومشاريع إنترنت الأشياء دون الحاجة إلى وحدات إضافية	لوحات Arduino القياسية لا تحتوي على Wi-Fi أو Bluetooth مدمجين. يتطلب إضافة الاتصال اللاسلكي دروعاً أو وحدات خارجية، مما قد يزيد من تعقيد وتكلفة المشاريع.
يتميز ESP32 بعدد من قنوات ADC بدقة تصل إلى 12 بت، مما يوفر قراءات تناظرية أكثر دقة ويسمح بمراقبة المدخلات التماثلية بشكل متزامن.	على سبيل المثال، يحتوي Arduino Uno على علمحول تماثلي رقمي بدقة 10 بت مع عدد أقل من القنوات، مما يوفر دقة أقل وخيارات أقل للقراءة الإشارات التناظرية بشكل متزامن
يتضمن ESP32 قناتين DAC مدمجتين، مما يتيح توليد إشارات تماثلية مباشرة من القيم الرقمية، وهو مفيد في تطبيقات الصوت والتحكم في الإخراج المتغير	معظم لوحات Arduino القياسية، مثل Arduino Uno، لا تحتوي على قدرات DAC مدمجة وتحتاج إلى مكونات إضافية لتوليد الإشارات التماثلية من الإشارات الرقمية.
يأتي مع كمية كبيرة من SRAM (520KB) بالإضافة إلى تخزين SPI فلاش إضافي، مما يسمح بتشغيل برامج أكبر وأكثر تعقيداً	يحتوي Arduino Uno : على سبيل المثال، على 2 KB من SRAM و 32 KB من ذاكرة فلاش، مما يحد من حجم وتعقيد البرامج التي يمكن تشغيلها

<p>يعمل بجهد V3.3، وهو شائع في الإلكترونيات الحديثة ويتطلب تحويل مستوى الجهد للتفاعل مع الحساسات والمكونات التي تعمل بجهد V5</p>	<p>العديد من لوحات Arduino، مثل Arduino Uno، تعمل بجهد V5، مما يجعلها متوافقة مع مجموعة واسعة من الحساسات والمكونات القديمة دون الحاجة إلى تحويل مستوى الجهد.</p>
<p>على معالج مزدوج النواة، مما يتيح تنفيذ المهام بشكل متوازٍ حقيقي ويعزز الكفاءة في تعدد المهام</p>	<p>تحتوي معظم لوحات Arduino، مثل Uno، على معالج أحادي النواة، مما يعني أنها يمكنها التعامل مع مهمة واحدة فقط في كل مرة، مما يجعل تعدد المهام أقل كفاءة وأكثر تعقيداً</p>
<p>يوفر عدداً أكبر من منافذ GPIO، والعديد منها متعدد الوظائف ويمكن تكوينه لاستخدامات متنوعة، بما في ذلك استشعار اللمس، وPWM، والمزيد [17].</p>	<p>عادةً ما تحتوي على عدد أقل من منافذ GPIO، ورغم أنها متعددة الاستخدامات، فإنها غالباً ما تفتقر إلى القدرات متعددة الوظائف التي توجد في منافذ ESP32</p>

#### 4.2 الشاشات البلورية السائلية (عارض LCD)

##### 1.4.2 تعريف

هي جهاز بصري يتم التحكم فيه إلكترونياً، يأخذ شكل لوحة رقيقة ومسطحة مكونة من عدد من البكسلات الملونة أو الأحادية اللون، المملوءة ببلورات سائلة ومصفوفة أمام مصدر ضوء (إضاءة خلفية) أو عاكس، تستخدم عادة في الأجهزة الإلكترونية التي تعمل بالبطاريات لأنها تستهلك كميات صغيرة جداً من الطاقة الكهربائية [20].



الشكل (12.2): شاشة LCD

## 2.4.2 البلورات السائلة

نعلم أن المواد في الطبيعة إما في الحالة الصلبة أو السائلة أو الغازية، فالحالة الصلبة تكون فيها جزيئات المادة مرتبة باتجاه محدد وفي مواقع محددة بالنسبة لبعضها البعض أي لا تتحرك، أما في الحالة السائلة فإن جزيئاتها تكون في حالة حركة مستمرة ولا يجمعها اتجاه ترتيب محدد، ولكن هناك بعض المواد تكون في حالة وسطية أي بين السائل والصلب حيث تحافظ جزيئات المادة في هذه الحالة على اتجاه ترتيبها كما في الحالة الصلبة ولكن في نفس الوقت تتحرك مثل جزيئات الحالة السائلة وهذا يعني أن البلورات السائلة هي ليست حالة صلبة وليست حالة سائلة ولكن بين الحالتين معا من هنا جاءت التسمية بالبلورات السائلة.

إذا هل يمكن اعتبار البلورات السائلة تتصرف مثل المواد الصلبة أو السائلة؟ في الحقيقة إنها أقرب إلى المواد السائلة منها إلى المواد الصلبة، باعتبار أن ارتفاع بسيط في الحرارة يحولها إلى سائلة ولهذا فإنها حساسة للتغيرات في درجة الحرارة [21].

## 3.4.2 المكونات الأساسية لشاشة LCD

- طبقة ضوء الخلفية (Backlight).

- طبقة مشتتة للضوء.

- مرشح خلفي.

- طبقة من الترانزستورات TFT [22].

## 4.4.2 كيف تعمل شاشات LCD

يظهر الضوء من خلف الشاشة إما عن طريق طبقة عاكسة في النمط الانعكاسي أو من منبع ضوئي مدمج في النمط المنقول، وتقوم طبقة الاستقطاب بترشيح الضوء، الذي يمر من خلال طبقة زجاجية، وطبقة إلكترونات شفافة، وطبقة تراصف، ومن ثم من خلال الكريستالات السائلة ذاتها، وتأتي بعدها طبقات متممة، مؤلفة من طبقة إلكترونات شفافة، وطبقة تراصف، ثم مرشحات الألوان وشريحة زجاجية، ويتألف كل بيكسل في الشاشة الملونة من جزء أحمر وآخر أخضر وثالث

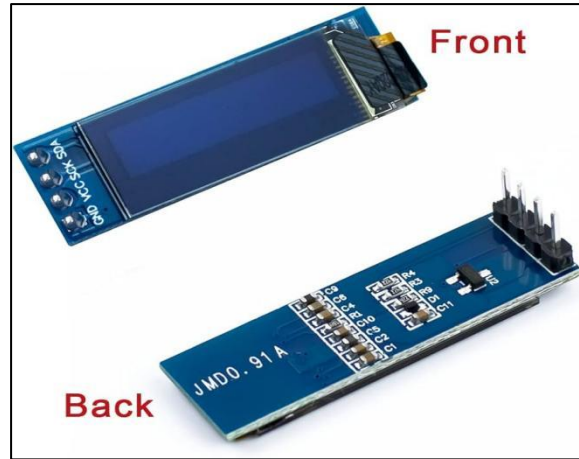
أزرق بنسب تحددها التعليمات التي تتحكم بالكريستالات السائلة عند كل نقطة على شبكة الإلكترودات الشفافة وإذا لم تكن المذبذبات السائلة مشحونة عند نقطة معينة فإن الضوء يمر عبر طبقة الاستقطاب الأمامية، وإذا لم تكن الكريستالات السائلة مشحونة كهربائياً، فإنها تمنع مرور الضوء.

يتم ترشيح الضوء عبر طبقة استقطاب وتنشئ طبقتا التراصف أشرطة من الكريستالات السائلة قضيبية الشكل والتي تتراصف بشكل طبيعي جنباً إلى جنب مسببة التواء هذه الأشرطة بزاوية قدرها 90 درجة، ويلتوي الضوء على طول أشرطة الكريستال السائل ويمر عبر طبقة استقطاب ثانية، متموضعة بانزياح قدره 90 درجة عن طبقة الاستقطاب الأولى.

إذا تلقت أشرطة الكريستال السائل الشحنة الكهربائية من طبقة من الإلكترودات فإن الجزيئات تتراصف بحيث نوضع النهاية الأولى مع النهاية الأخرى، ما يسمح للضوء بالمرور بشكل مباشر بدون التواء، وفي تلك الحالة يمنع مرشح الاستقطاب الثاني مرور الضوء [21].

#### 4.2 لمحة عن تكنولوجيا OLED (Organic Light Emitting Diode)

أجهزة OLED مصنعة من مواد ذات أساس كربوني عضوي، تصدر ضوء عند تطبيق التيار الكهربائي عليها، وهي أكثر فعالية وبساطة من حيث الاستعمال من شاشات LCD والتي لا تعتمد على إضاءة الخلفية ولا الفلاتر، وتؤمن جودة صورة جيدة بوضوح عالٍ، كذلك لها مميزات لونية رائعة، كما تملك استجابة سريعة نسبياً ومعدلاً أوسع من زوايا الرؤية، وتستعمل أيضاً لصناعة أضواء الـ OLED نشأت هذه التكنولوجيا في بداية 1980، وطوّرت بشكل كبير لتستبدل تقنية الـ LCD لأنّ تكنولوجيا الـ OLED أكثر سطوعاً وأرق وأخف مقارنة بالـ LCD، وتستهلك طاقة أقل من الـ LCD وتوفّر مميزات تباين أكبر، فالفائدة الأكثر اجتذاباً التي تملكها هذه التكنولوجيا بالنسبة لـ LCD هي أنّها أرخص في التصنيع وبالتالي أكثر فعالية في التكلفة [23].



الشكل (13.2): شاشة OLED 0.91"

## 5.2 خاتمة

تناولنا في هذا الفصل تجهيزات المشروع بشكل عام، وسيتم التوضيح والتفصيل في الفصل الموالي لتقنية ونوع لوحات التطوير المستعملة التي تساعد في تنفيذ المشروع بشكل مثالي.

---

الفصل الثالث  
دراسة تطبيقية للمشروع

---

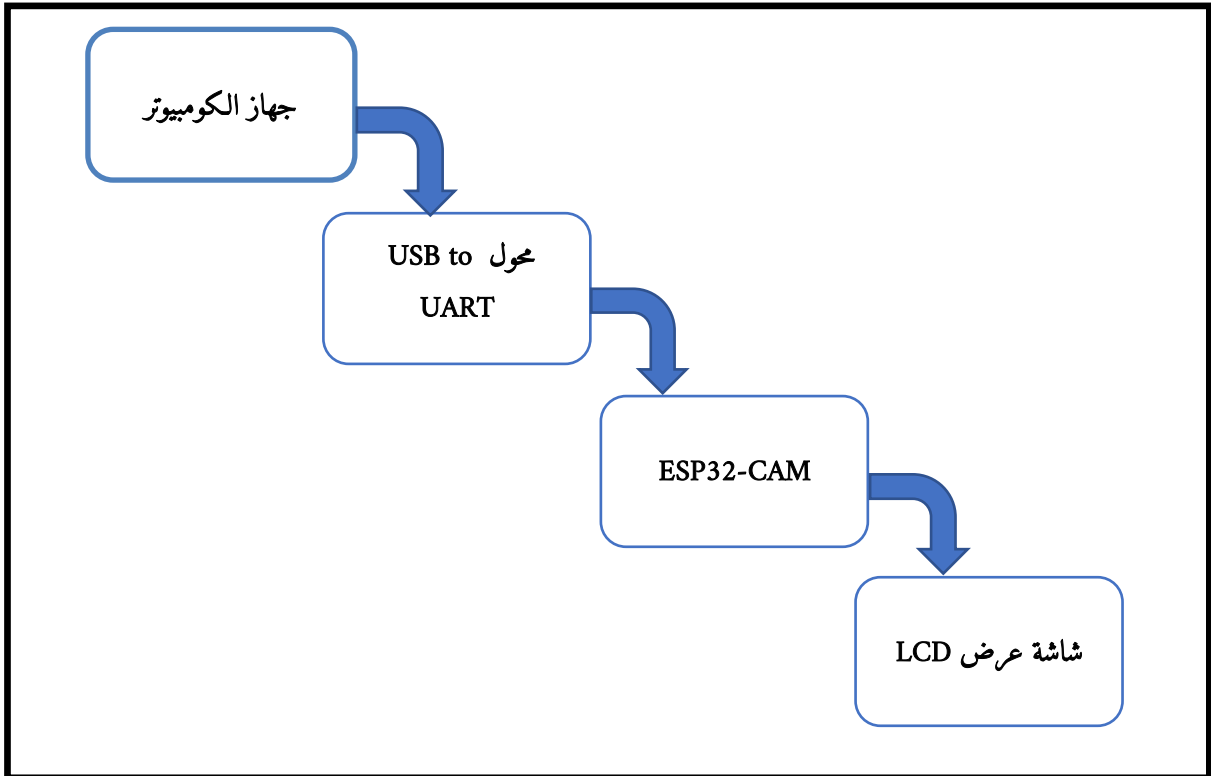
## 1.3 مقدمة

بعد الدراسة النظرية لمختلف العناصر الالكترونية الضرورية لانجاز بطاقة الكترونية للتعرف على الصور سنتناول في هذا الفصل مراحل تطبيق هذا المشروع ثم محاكاته لتسهيل انجاز العمل التطبيقي ومن ثم تطبيقه على أرض الواقع.

## 2.3 المخطط الصندوقي

لتمثيل البنية العامة لنظام التعرف على الوجوه باستخدام وحدة ESP32-CAM ، تم اعتماد مخطط صندوقي يوضح المكونات الرئيسية للنظام والعلاقات بينها. يُعتبر المخطط الصندوقي أداة هندسية تساعد على تبسيط فهم النظام، من خلال عرض كل مكون على شكل "صندوق" مع تحديد دور كل مكون وتدفق البيانات أو الإشارات داخله.

في هذا المشروع، يوضح المخطط الصندوقي كيفية تفاعل الأجهزة والبرمجيات معاً لتحقيق وظيفة التعرف على الوجوه، انطلاقاً من التقاط الصورة إلى عرض اسم الشخص المتعرف عليه.



الشكل (1.3): المخطط الصندوقي لانجاز نظام التعرف على الوجه بواسطة ESP32-CAM

## 3.3 وصف النظام

يتكوّن النظام المقترح من منصة تعتمد على وحدة ESP32-CAM، وهي لوحة إلكترونية مدمجة تتضمن كاميرا رقمية ومعالجاً قادراً على معالجة الصور. يهدف هذا النظام إلى التعرف على وجوه الأشخاص في الزمن الحقيقي، من خلال التقاط صورة للوجه باستخدام الكاميرا، ثم تحليلها ومقارنتها بقاعدة بيانات داخلية.

يُبرج النظام انطلاقاً من جهاز الكمبيوتر عبر بيئة تطوير مثل Arduino IDE، وذلك باستخدام كابل محول USB إلى UART لتمكين الاتصال التسلسلي مع وحدة ESP32-CAM. بعد تحميل البرنامج، يمكن للوحدة العمل بشكل مستقل.

عند تشغيل النظام، تقوم وحدة ESP32-CAM بالتقاط صورة للوجه الموجود أمام الكاميرا، ثم تمررها إلى خوارزمية مدمجة للتعرف على الوجه غالباً ضمن مكتبة (ESP-WHO) إذا تم التعرف على الشخص، يقوم المعالج بعرض اسم الشخص المتعرف عليه على شاشة خارجية مرتبطة بالوحدة. هذا النظام يُمكن استخدامه في تطبيقات متعددة، مثل: أنظمة الحضور الذكي، التحكم في الوصول، والمراقبة الذكية.

## 4.3 العناصر الإلكترونية المستعملة

## ○ لوحة ESP32-CAM

بطاقة ESP32-CAM هي بطاقة تطوير قوية ومنخفضة التكلفة تعتمد على المتحكم ESP32 من شركة Espressif Systems، وتتميز بكاميرا مدمجة من نوع OV2640 بدقة تصل إلى 2 ميغابكسل، مما يجعلها مثالية لمشاريع المراقبة والتعرف على الصور. يمكن برمجتها باستخدام Arduino IDE أو ESP-IDF، وتحتوي على منافذ متعددة مثل GPIO و UART و I2C و SPI لتوصيل مختلف الحساسات والمحركات. من المهم ملاحظة أن البطاقة لا تحتوي على محول تماثلي-رقمي مدمج، لذا نطلب وحدة خارجية لقراءة الإشارات التماثلية، كما قد تحتاج إلى مصدر طاقة خارجي لتشغيل الكاميرا بسبب محدودية التيار الذي يوفره المتحكم.



الشكل (2.3): بطاقة ESP32-CAM

### ○ شاشة oled (organic Light- Emitting Diode)

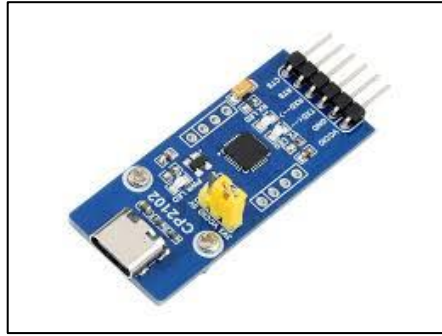
- شاشة عرض "0.91" بدقة 32×128 واجهة I2C باللون الأبيض
- شاشات OLED تتحكم بكل بكسل بشكل منفصل وتصدر ضوءها الخاص على عكس شاشات LCD التي تعتمد على إضاءة خلفية توفر جودة عرض ممتازة مع ألوان زاهية وحركة سريعة وتباين عالي.
- أداء عرض أفضل من شاشات LCD التقليدية.
- تعمل عبر واجهة I2C القياسية.
- لا تحتاج لإضاءة خلفية (إضاءة ذاتية).
- استهلاك طاقة أقل.
- شريحة تحكم من نوع SSD1306
- حجم الشاشة "0.9"
- جهد تشغيل من 3.3 الى 5 فولت
- دقة عرض 32 × 128 بكسل



الشكل (3.3): شاشة OLED "0,91"

## ○ محول USB to UART:

يُستخدم محول USB إلى UART لربط الحاسوب بالأجهزة الإلكترونية التي تعتمد على الاتصال التسلسلي (مثل المتحكمات الدقيقة) عبر منفذ USB. يقوم هذا المحول بتحويل إشارات USB إلى إشارات TTL متوافقة مع بروتوكول UART، ما يسمح ببرمجة المتحكمات ونقل البيانات وتصحيح الأخطاء. من أشهر شرائحه: CH340، FT232RL، وCP2102، ويحتوي عادةً على منافذ TX، RX، وGND، وأحياناً V<sub>CC</sub>. يُعد هذا المحول أداة ضرورية في مشاريع الأنظمة المدججة، خاصة عند غياب المنفذ التسلسلي في الحاسوب



الشكل (4.3): محول من نوع CH340G

## 5.3 تحضير قاعدة البيانات

قبل الشروع في العمل التطبيقي يجب علينا انشاء قاعدة بيانات خاصة بالصورة المراد التعرف عليها لادراجها في الكود المعمول به حيث يتم رفع الصور الى الموقع [/https://edgeimpulse.com](https://edgeimpulse.com) يجب مراعاة أن تكون الصور مناسبة بدون خلفية أو على الأق بخلفية ذات لون واحد وكلها زاد عدد الصور زادت دقة النموذج.

يتم تحضير قاعدة البيانات حسب الخطوات التالية:

## الخطوة الأولى: تسجيل الدخول

انشاء حساب على Edge Impulse أو اذا كان لدينا حساب مسبقا نقوم بالدخول.

الشكل (5.3): تسجيل الدخول

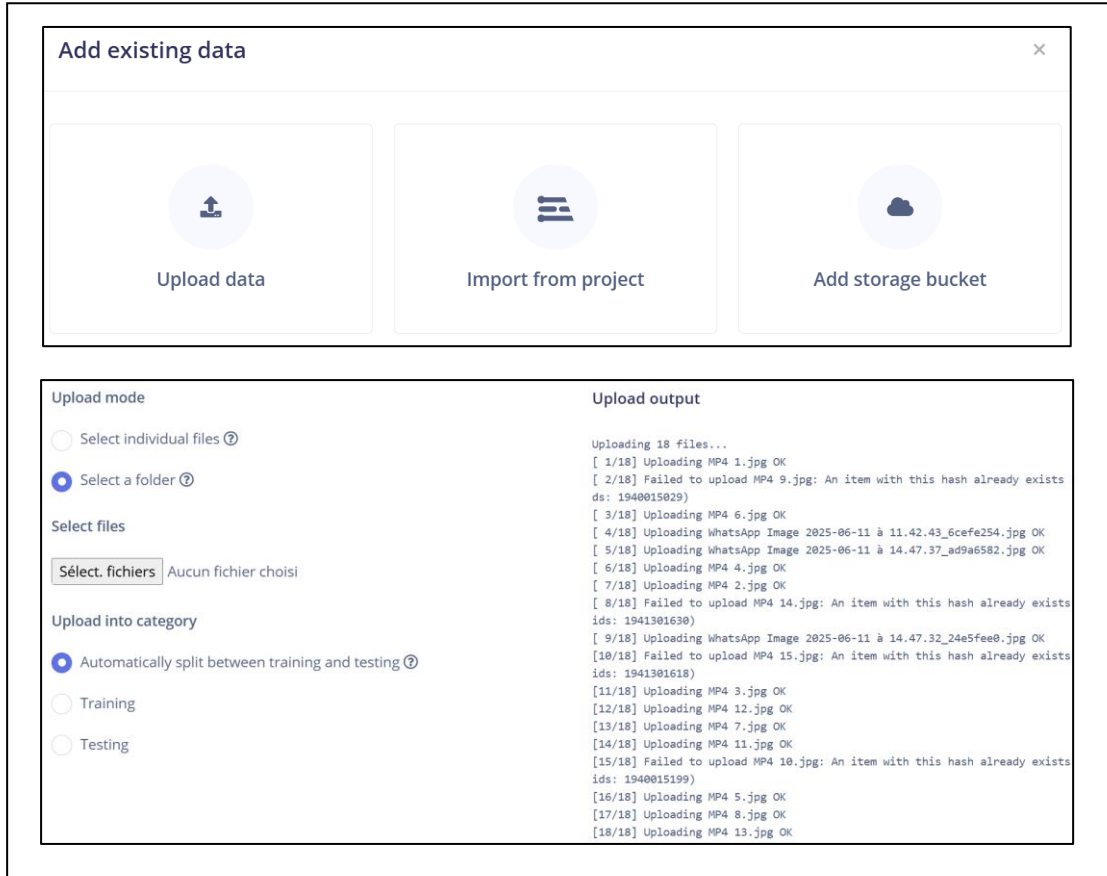
الخطوة الثانية: إنشاء مشروع جديد

وذلك بالضغط على add existing data.

الشكل (6.3): إنشاء مشروع جديد

الخطوة الثالثة: جمع البيانات Data Acquisition

في هذه الخطوة نقوم بإضافة البيانات التي قمنا بجمعها مسبقا وذلك من خلال قائمة Data Acquisition، يمكن رفع الصور على شكل فردي أو كمجلد كامل الصيغ المقبولة للملفات هي: .CSV,JSON,AVI,CBOR,JPG,PNG,MP4,WAV.

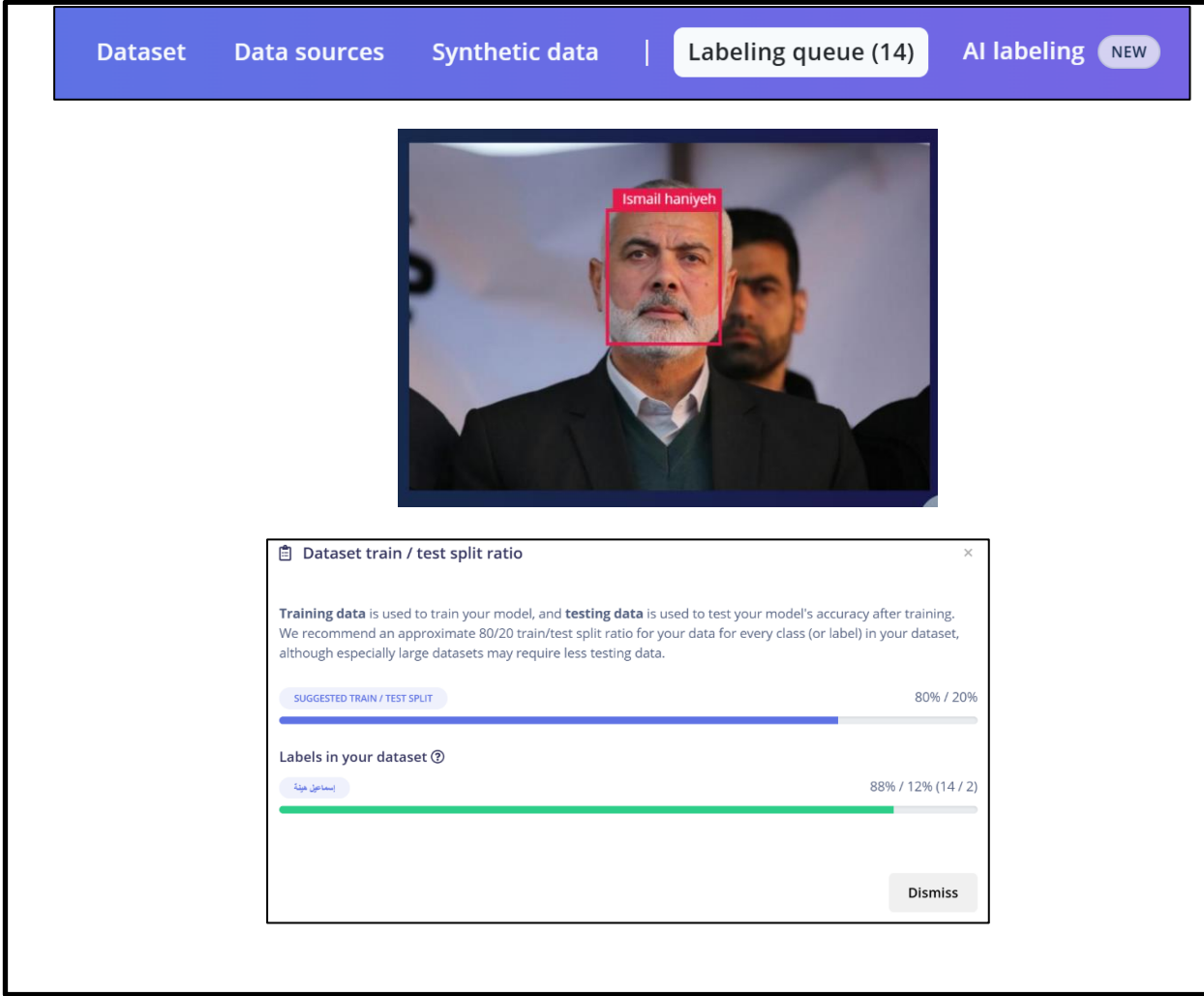


الشكل (7.3): جمع البيانات

### الخطوة الرابعة:

#### 1. تسمية البيانات:

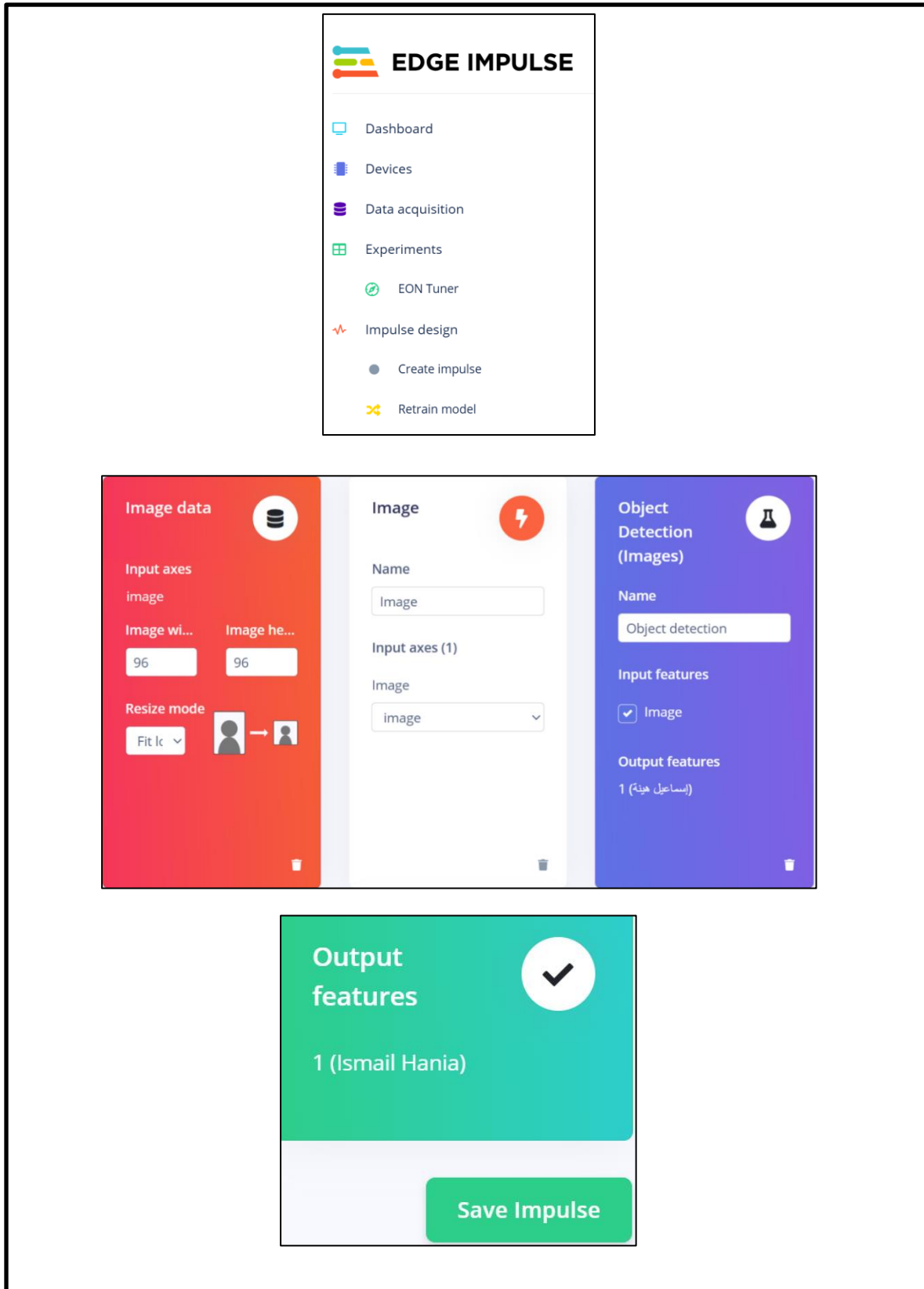
من خلال النقر على Labeling Queue هنا نحتاج الى رسم مربع احاطة Bounding Box حول الوجه في هذه الصورة ونقوم بتسميته، بعد الإنتهاء تظهر نسبة التقسيم (تدريب: اختبار) مثل 79:21 قربية من النسبة المثالية 80:20.



الشكل (8.3): تسمية البيانات

## 2. إنشاء نبضة (Impulse):

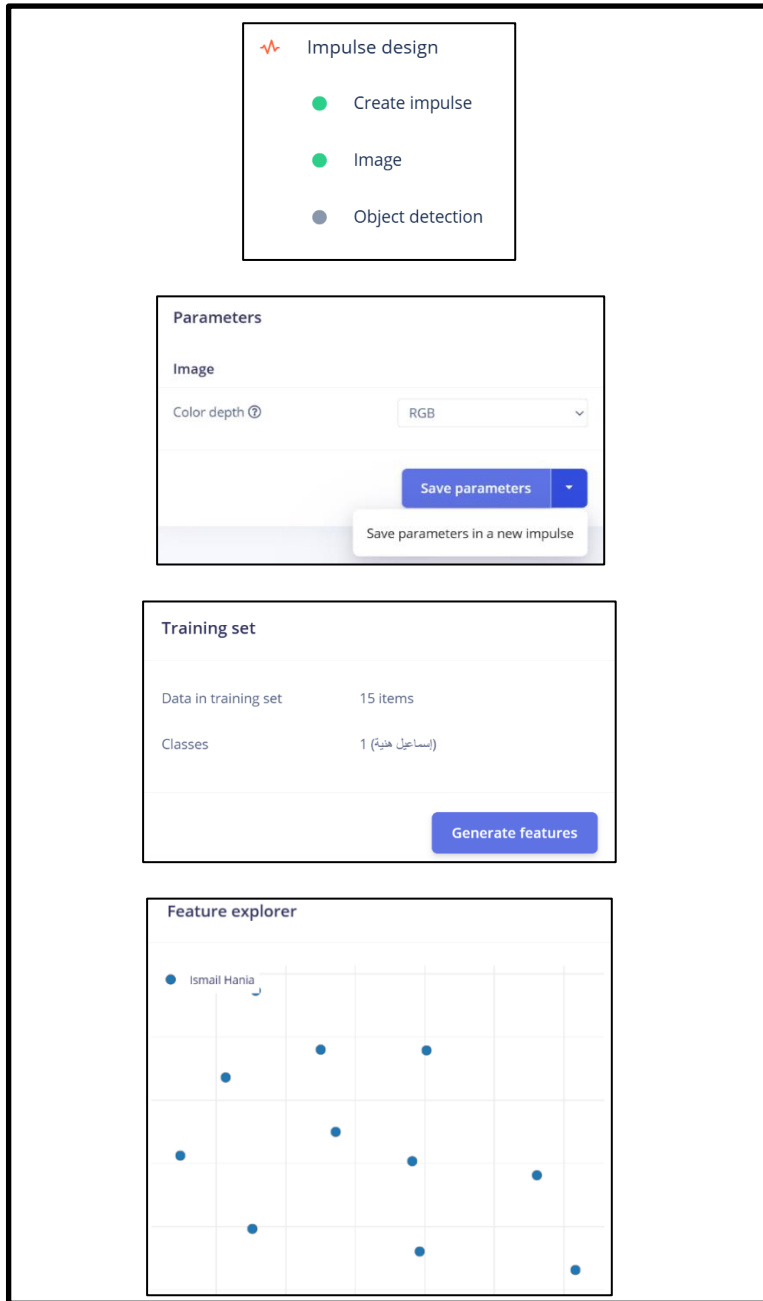
- نذهب إلى Create Impulse ثم Image Data، يفضل استخدام حجم 96x96 بكسل (يمكن تغييره لكن الحجم الأكبر يزيد من وقت الاستدلال).
- ثم نختار وضع تغيير الحجم (يفضل fit longest Axis) ثم نضيف كتلة المعالجة وكتلة المعالجة وكتلة التعلم (processing block \ learning block).
- نتحقق من ميزة الاخراج (output features) للتأكد من أنها تلي المتطلبات وأخيرا نقوم بحفظ النبضة (Save impulse).



الشكل (9.3): إنشاء نبضة

### 3. توليد المزايا (generating feature):

ضمن نفس القائمة ننتقل الى خيار Image ونحدد عمق اللون يفضل تدرج الرمادي لتقليل المعالجة او نختار RGB إن لزم، ثم نقر على حفظ الاعدادات فيتم توجيهنا الى نافذة generate features ونقر على زر لتوليد المزايا، ومن ثم سيتم انشاء مستعرض الميزات feature explorer (إذا كانت الكرات الممثلة للتصنيفات متجمعة حسب اللون يدل هذا على الجودة الجيدة للبيانات).



الشكل (10.3): توليد المزايا

الخطوة الخامسة:

1. تدريب النموذج:

○ ننتقل الى خيار object detection ونترك الاعدادات كما هي فقط نغير معدل التعلم learning rate الى 0.01.

○ نضغط على start training لبدء التدريب، بعد الانتهاء ستظهر نسبة الدقة f1 score (نحاول ان تكون أعلى من 80%).

The screenshot displays the 'Impulse design' training interface. It includes a 'Training settings' section with the following parameters: Number of training cycles (60), Use learned optimizer (unchecked), Learning rate (0.01), Training processor (CPU), and Data augmentation (checked). Below this is the 'Neural network architecture' section, showing an 'Input layer (27,648 features)'. At the bottom, the 'F1 SCORE' is displayed as 100.0%. A 'Confusion matrix (validation set)' is also shown, with the following data:

	BACKGROUND	ISMAIL HANIA
BACKGROUND	100%	0%
ISMAIL HANIA	0%	100%
F1 SCORE	1.00	1.00

الشكل (11.3): تدريب النموذج

## 2. نشر النموذج الى ESP32-CAM:

- نذهب الى deployment ثم نختار نوع النشر (Arduino library).
- نختار نوع اللوحة Expressif ESP-EYE (ESP32 240MHZ).
- نضغط على build وسيتم تحميل ملف Zip فيه مكتبة النموذج.

The screenshot displays a deployment interface with the following components:

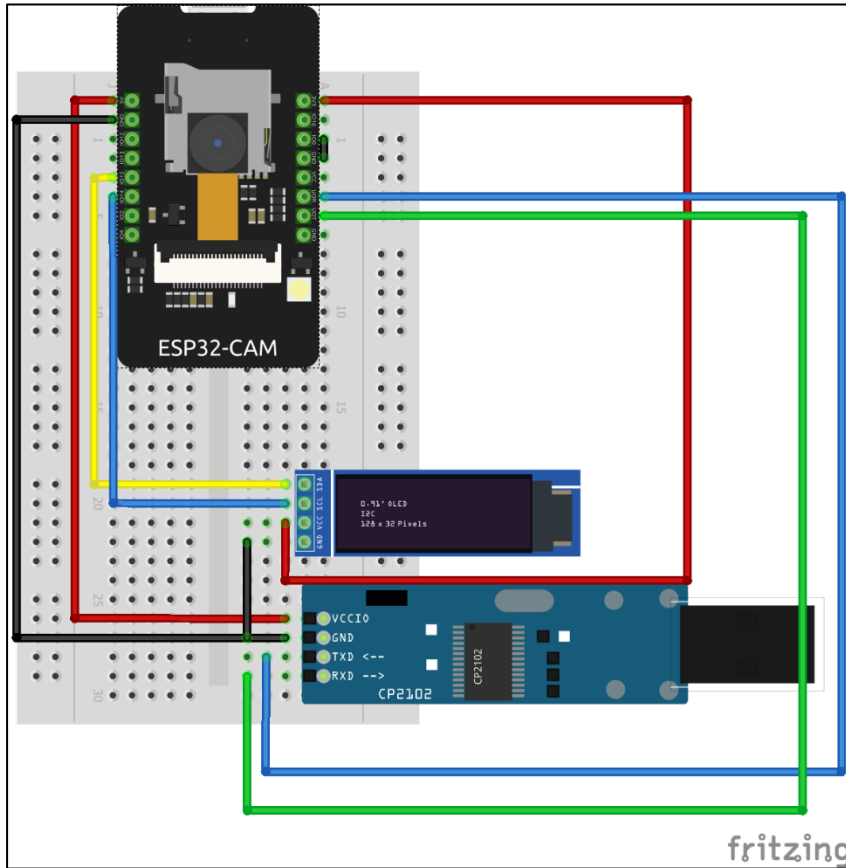
- Deployment Options:** A list of actions: Retrain model, Live classification, Model testing, and Deployment.
- Search Deployment Options:** A search bar with the text "search deployment options". Below it, three results are shown:
  - C++ library:** A portable C++ library with no external dependencies, which can be compiled with any modern C++ compiler.
  - Arduino library:** An Arduino library with examples that runs on most Arm-based Arduino development boards.
  - BrainChip MetaTF Model for Akida - AKD1000 and AKD1500:** A MetaTF converted model (.fbz) for use with the BrainChip Akida™ runtime.
- Target device configuration:** A section titled "Target device" with the instruction: "Define your target device requirements to inform model optimizations and performance calculations. No device yet? Use the default settings which you can change at any time." It includes three dropdown menus:
  - Target device:** Espressif ESP-EYE (ESP32 240MHz)
  - Processor family:** ESP32
  - Clock rate:** 240 MHz (with a "Max" label below it)
- Success Message:** A green checkmark icon followed by the text "Built Arduino library". Below this, it says: "Add this library through the Arduino IDE via: Sketch > Include Library > Add .ZIP Library...". It also provides the path: "Examples can then be found under: File > Examples > a202034018920-project-1\_inferencing".

الشكل (12.3): رفع النموذج

## 6.3 محاكاة المشروع

برنامج Fritzing هو برنامج مفتوح المصدر يُستخدم في تصميم الدارات الإلكترونية بطريقة مرئية ومبسطة، ويُعتبر أداة تعليمية فعّالة للمبتدئين والهواة وحتى للطلبة. يوفر البرنامج بيئة عمل تتيح رسم مخططات الدارات على لوحة التجرب (Breadboard)، وتحويلها إلى مخططات إلكترونية احترافية (Schematic) وحتى تصميم الدارات المطبوعة (PCB).

اخترنا استعمال برنامج Fritzing في هذا المشروع من أجل تمثيل ومحاكاة التوصيلات الإلكترونية، خاصة وأنه يدعم مجموعة واسعة من المكونات الحديثة مثل لوحة ESP32-CAM، التي لا تتوفر في برامج محاكاة أخرى مثل Proteus، حيث لا يدعم هذا الأخير هذه الوحدة بشكل مباشر. لذلك، وفر لنا Fritzing حلاً مناسباً لتصميم التوصيلات والتحقق منها بصرياً قبل التطبيق العملي.



الشكل (13.3): محاكاة المشروع باستعمال برنامج fritzing

### 7.3 شرح الدارة

قنا في دارة بتوصيل كالتالي:

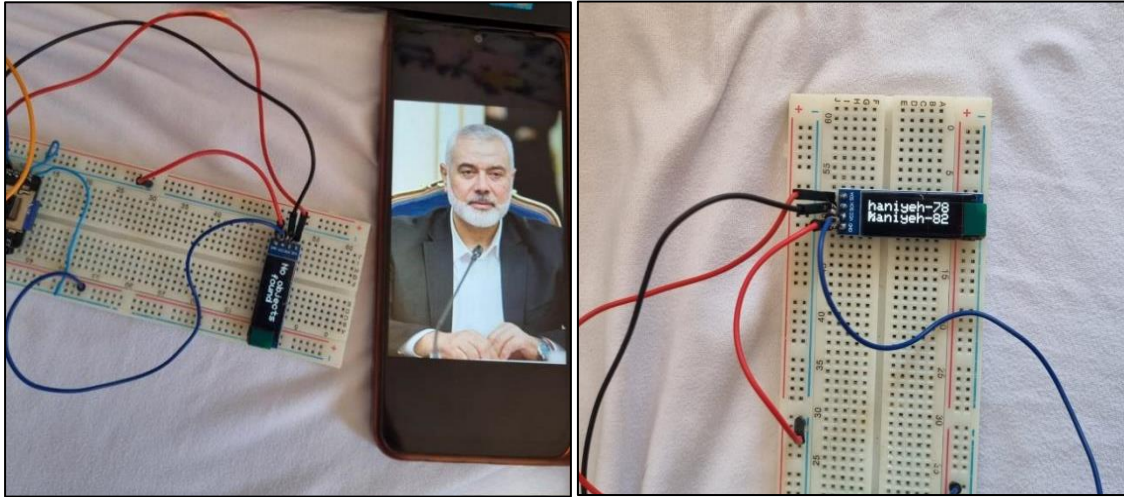
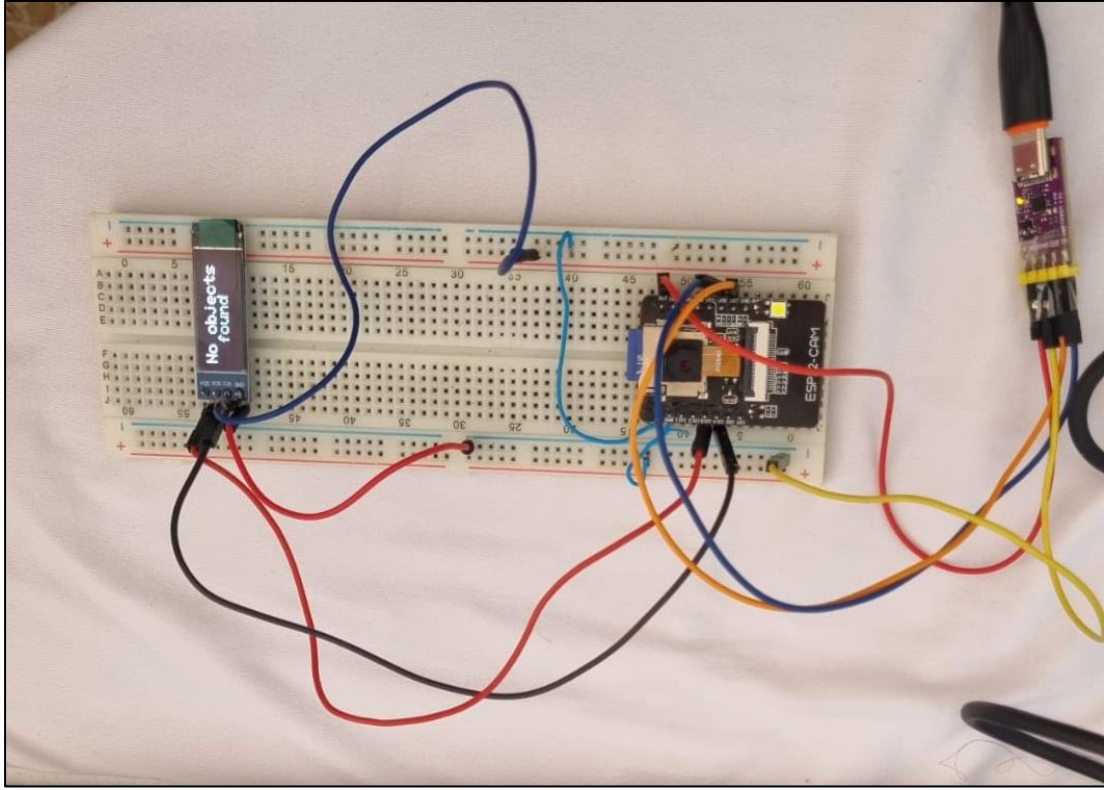
الجدول (1.3): توصيل لوحة ESP32-CAM مع المحول

ESP32-CAM	المحول
5V	Vcc
GND	GND
UOR	TX
UOT	RX

الجدول (2.3): توصيل لوحة ESP32-CAM مع شاشة OLED

ESP32-CAM	شاشة OLED
3.3v	Vcc
GND	GND
IO14	SCL
IO15	SDA

## 8.3 صور حقيقية للمشروع



الشكل (14.3): صور حقيقة للمشروع

## 9.3 خاتمة

في الأخير، تحصلنا على النتيجة المرجوة حيث تمكنا انجاز نظام للتعرف على الوجه باستعمال لوحة ESP32-CAM، فبعد ادخال بيانات الصورة المتعرف عليها يقوم النظام بالتعرف على الوجه تلقائيا مع عرض الاسم الموافق له.

---

خاتمة عامة

---

## خاتمة عامة

في ختام هذه الدراسة، يمكننا التأكيد على أن تقنية التعرف على الصور والوجوه باستخدام وحدة ESP32 CAM تشكل خطوة هامة نحو تطوير الأنظمة الذكية والابتكار التكنولوجي في العديد من المجالات. من خلال استعراض المكونات الأساسية والتقنيات المتبعة في هذا المجال، تمكنا من تصميم نظام مرن وسهل البرمجة باستخدام بيئة Arduino IDE، مما ساعد في تسريع عملية التطوير وتحقيق تطبيقات عملية للذكاء الاصطناعي في الحياة اليومية.

لقد بدأنا دراستنا بتناول المفاهيم الأساسية المتعلقة بالصور الرقمية ومعالجتها، ودرسنا بعدها وحدة ESP32 CAM وخصائصها التي تجعلها أداة مثالية لهذا النوع من التطبيقات. كما تطرقنا إلى بيئة Arduino، التي ساعدت في تسهيل وتسريع عملية البرمجة.

من خلال التطبيق العملي لهذا المشروع، اكتشفنا أن تطوير الأنظمة الذكية لا يقتصر فقط على التطبيقات التقنية، بل هو عملية مستمرة من البحث والتطوير تهدف إلى تحسين الأداء وتوسيع التطبيقات الممكنة. العمل على هذا المشروع منحنا فهماً أعمق لكيفية تكامل مكونات النظام بشكل دقيق، مما عزز قدرتنا على التعامل مع التقنيات الحديثة بأسلوب علمي.

نأمل أن تكون هذه الدراسة قد قدمت إضافة حقيقية للمجال، وأن تساهم في تحفيز الباحثين والمطورين على استكشاف هذه التكنولوجيا وتوسيع نطاق تطبيقاتها. كما نرجو أن يكون هذا العمل قد حقق الهدف المنشود منه، وأن يشكل نقطة انطلاق لمزيد من البحث والتطوير في المستقبل.

---

## قائمة المراجع

---

## المراجع

[1] أ. د / محمد جلال علي، د / أمل يوسف عبد المجيد، د / محمود فوزي أنور، مريهان محي الدين محمد محمد "تطور مفهوم الصورة في ضوء الفنون الرقمية" جويلية 2023.

[2] Rafael C. Gonzalez • Richard E. Woods, "Digital Image Processing

Global Edition, 330 Hudson Street, New York, NY 10013

,University of Tennessee.

[3] Chrissolomon/toby breckon, "FUNDAMENTALS OF DIGITAL

IMAGE PROCESSING", University of California, Davis..

[4] م. كزار ضياء محمد، "تقنيات معالجة الصور الرقمية"، جامعة الكوفة / كلية علوم الحاسوب والرياضيات / قسم الرياضيات.

[5] هند رستم، محمد شعبان، "أساسيات معالجة الصور الرقمية"، رقم الابداع في دار الكتب والوثائق الوطنية ببغداد (1058) لسنة 2008.

[6] <https://www.britannica.com/technology/pixel>

اطلع عليه يوم 9 جانفي 2025

[7] [https://www.noor-book.com/tag/%D8%B9%D9%85%D9%82-](https://www.noor-book.com/tag/%D8%B9%D9%85%D9%82-%D9%84%D9%88%D9%86%D9%8A)

%D9%84%D9%88%D9%86%D9%8A

اطلع عليه يوم 9 جانفي 2025

[8] Aditya Kumar, "What Is Image Processing : Overview, Applications

Benefits, and More", Artificial Intelligence Engineer, 2024.

[9] <https://smartcodewave.com/image-processing/>

اطلع عليه يوم 12 جانفي 2025

[10] Tinku acharya, ajoy K.Ray, "Image processing principales and

applications", WILEY-INTERSCIENCE, A JOHN WILEY & SONS, MC.,

PUBLICATION

[11] حسام الوفاي، "الأردوينو من البداية الى الاحتراف" (مستوى المبتدئ 2020)

[12] عبد الله علي عبد الله، "أردوينو ببساطة" 3 أوت 2012.

[13] محمود المسلماني "الأردوينو كما لم تعرفه من قبل" مكتبة أمجاد 23 سبتمبر 2019

[14] فريق عمل القرية الهندسية "كتاب احتراف الأردوينو في عشر أيام" مؤسسة القرية الهندسية،

الطبعة الثانية 2015

[15] منتدى القرية الالكترونية بيئة التطوير المتكاملو للأردوينو .

[16] أحمد أفغاني "كتاب أساسيات الأردوينو"، electro ahmed 14 ديسمبر 2018.

[17] <https://www.arduinomaster.com/2024/06/esp32.html>

أطلع عليه يوم 22 ديسمبر 2024.

[18] <https://www.robotique.site/tutoriel>

arabe/%D9%84%D9%88%D8%AD%D8%A9-esp32/

أطلع عليه يوم 22 ديسمبر 2024

[19] <https://github.com/yellobyte/ESP32-DevBoards-Getting-Started.git>

أطلع عليه يوم 22 ديسمبر 2024.

[20] "Liquid Crystal Displays: A Tutorial" John L. McCann

[20] علاء الدين عبد المنعم "شاشات LCD" 17 أوت 2008

[21] محمد حميد حسين "شاشات العرض LCD" كلية التربية للعلوم الصرفة-قسم الفيزياء

تم الاطلاع 27 ديسمبر 2024

[22] Sarah Zarein Raza, "Understanding OLED Display Technology", Circuit digest, Published July 13, 2015.

[23] International Journal of Engineering And Computer Science ISSN:2319-7242 Volume1 Page No. 75-84

---

الملاحق

---

```
#include <a202034018920-project-1_inferencing.h>
#include "edge-impulse-sdk/dsp/image/image.hpp"
#include "esp_camera.h"
#include <Wire.h>
#include <Adafruit_GFX.h>
#include <Adafruit_SSD1306.h>
#define I2C_SDA 15
#define I2C_SCL 14
TwoWire I2Cbus = TwoWire(0);
#define SCREEN_WIDTH 128
#define SCREEN_HEIGHT 32
#define OLED_RESET -1
#define SCREEN_ADDRESS 0x3C
Adafruit_SSD1306 display(SCREEN_WIDTH, SCREEN_HEIGHT, &I2Cbus,
OLED_RESET);
#define CAMERA_MODEL_AI_THINKER #if defined(CAMERA_MODEL_ESP_EYE)
#define PWDN_GPIO_NUM
#define RESET_GPIO_NUM
#define XCLK_GPIO_NUM
#define SIOD_GPIO_NUM
#define SIOC_GPIO_NUM
#define Y9_GPIO_NUM
#define Y8_GPIO_NUM
#define Y7_GPIO_NUM
#define Y6_GPIO_NUM
#define Y5_GPIO_NUM
#define Y4_GPIO_NUM
#define Y3_GPIO_NUM
#define Y2_GPIO_NUM
#define VSYNC_GPIO_NUM
#define HREF_GPIO_NUM
#define PCLK_GPIO_NUM
#elif defined(CAMERA_MODEL_AI_THINKER)
#define PWDN_GPIO_NUM
#define RESET_GPIO_NUM
#define XCLK_GPIO_NUM
#define SIOD_GPIO_NUM
#define SIOC_GPIO_NUM
#define Y9_GPIO_NUM
```

```

#define Y8_GPIO_NUM
#define Y7_GPIO_NUM
#define Y6_GPIO_NUM
#define Y5_GPIO_NUM
#define Y4_GPIO_NUM
#define Y3_GPIO_NUM
#define Y2_GPIO_NUM
#define VSYNC_GPIO_NUM
#define HREF_GPIO_NUM
#define PCLK_GPIO_NUM
#else
#error "Camera model not selected"
#endif
#define EI_CAMERA_RAW_FRAME_BUFFER_COLS
#define EI_CAMERA_RAW_FRAME_BUFFER_ROWS
#define EI_CAMERA_FRAME_BYTE_SIZE
static bool debug_nn = false;
static bool is_initialised = false;
uint8_t *snapshot_buf;
static camera_config_t camera_config = {
    .pin_pwdn = PWDN_GPIO_NUM,
    .pin_reset = RESET_GPIO_NUM,
    .pin_xclk = XCLK_GPIO_NUM,
    .pin_sscb_sda = SIOD_GPIO_NUM,
    .pin_sscb_scl = SIOC_GPIO_NUM,
    .pin_d7 = Y9_GPIO_NUM,
    .pin_d6 = Y8_GPIO_NUM,
    .pin_d5 = Y7_GPIO_NUM,
    .pin_d4 = Y6_GPIO_NUM,
    .pin_d3 = Y5_GPIO_NUM,
    .pin_d2 = Y4_GPIO_NUM,
    .pin_d1 = Y3_GPIO_NUM,
    .pin_d0 = Y2_GPIO_NUM,
    .pin_vsync = VSYNC_GPIO_NUM,
    .pin_href = HREF_GPIO_NUM,
    .pin_pclk = PCLK_GPIO_NUM,
    .xclk_freq_hz = 20000000,
    .ledc_timer = LEDC_TIMER_0,
    .ledc_channel = LEDC_CHANNEL_0,
    .pixel_format = PIXFORMAT_JPEG, //YUV422, GRAYSCALE, RGB565, JPEG
    .frame_size = FRAMESIZE_QVGA,

```

```

    .jpeg_quality = 12,
      .fb_count = 1,
    .fb_location = CAMERA_FB_IN_PSRAM,
    .grab_mode = CAMERA_GRAB_WHEN_EMPTY,
};
bool ei_camera_init(void);
void ei_camera_deinit(void);
bool ei_camera_capture(uint32_t img_width, uint32_t img_height,
uint8_t *out_buf) ;
@brief
void setup()
{
    I2Cbus.begin(I2C_SDA, I2C_SCL, 100000);
    if (!display.begin(SSD1306_SWITCHCAPVCC, SCREEN_ADDRESS)) {
        Serial.printf("SSD1306 OLED display failed to
initialize.\nCheck that display SDA is connected to pin %d and SCL
connected to pin %d\n", I2C_SDA, I2C_SCL);
        while (true) ; }
    Serial.begin(115200);
    while (!Serial);
    Serial.println("Edge Impulse Inferencing Demo");
    if (ei_camera_init() == false) {
        ei_printf("Failed to initialize Camera!\r\n");
    }
    else {
        ei_printf("Camera initialized\r\n");
    }
    ei_printf("\nStarting continious inference in 2
seconds...\n");
    ei_sleep(2000);
    display.clearDisplay();
    display.setCursor(0, 0);
    display.setTextSize(1);
    display.setTextColor(SSD1306_WHITE);
    display.print("Starting continious\n inference in\n 2
seconds...");
    display.display();
    ei_sleep(2000);
    display.clearDisplay();
}
@brief      Get data and run inferencing

```

```

    @param[in] debug Get debug info if true
void loop()
{
display.clearDisplay();
    if (ei_sleep(5) != EI_IMPULSE_OK) {
        return;}
    snapshot_buf =
(uint8_t*)malloc(EI_CAMERA_RAW_FRAME_BUFFER_COLS *
EI_CAMERA_RAW_FRAME_BUFFER_ROWS * EI_CAMERA_FRAME_BYTE_SIZE);
    if(snapshot_buf == nullptr) {
        ei_printf("ERR: Failed to allocate snapshot buffer!\n");
        return; }
    ei::signal_t signal;
    signal.total_length = EI_CLASSIFIER_INPUT_WIDTH *
EI_CLASSIFIER_INPUT_HEIGHT;
    signal.get_data = &ei_camera_get_data;
    if (ei_camera_capture((size_t)EI_CLASSIFIER_INPUT_WIDTH,
(size_t)EI_CLASSIFIER_INPUT_HEIGHT, snapshot_buf) == false) {
        ei_printf("Failed to capture image\r\n");
        free(snapshot_buf);
        return;
    }
    ei_impulse_result_t result = { 0 };
    EI_IMPULSE_ERROR err = run_classifier(&signal, &result,
debug_nn);
    if (err != EI_IMPULSE_OK) {
        ei_printf("ERR: Failed to run classifier (%d)\n", err);
        return;
    }
    ei_printf("Predictions (DSP: %d ms., Classification: %d ms.,
Anomaly: %d ms.): \n",
                result.timing.dsp, result.timing.classification,
result.timing.anomaly);
#ifdef EI_CLASSIFIER_OBJECT_DETECTION == 1
    ei_printf("Object detection bounding boxes:\r\n");
    for (uint32_t i = 0; i < result.bounding_boxes_count; i++) {
        ei_impulse_result_bounding_box_t bb =
result.bounding_boxes[i];
        if (bb.value == 0) {
            continue;
        }
}

```

```

        ei_printf(" %s (%f) [ x: %u, y: %u, width: %u, height: %u
]\r\n",
                bb.label,
                bb.value,
                bb.x,
                bb.y,
                bb.width,
                bb.height);
    }
ei_printf("Object detection bounding boxes:\r\n");
bool bb_found = result.bounding_boxes[0].value > 0;
for (uint32_t i = 0; i < result.bounding_boxes_count; i++) {
    ei_impulse_result_bounding_box_t bb =
result.bounding_boxes[i];
    if (bb.value == 0) {
        continue;
    }
    ei_printf(" %s (%f) [ x: %u, y: %u, width: %u, height: %u
]\r\n",
            bb.label,
            bb.value,
            bb.x,
            bb.y,
            bb.width,
            bb.height);

    display.setCursor(0, 16 * i);
    display.setTextSize(2);
    display.setTextColor(SSD1306_WHITE);
    if (strcmp(bb.label, "Ismail haniyeh") == 0) {
        display.print("haniyeh");
    }
    display.print("-");
    display.print(int((bb.value) * 100));
    display.print("%");
    display.display();
}
if (!bb_found) {
    ei_printf(" No objects found\n");
    display.setCursor(0, 2);
    display.setTextSize(2);

```

```

        display.setTextColor(SSD1306_WHITE);
        display.print("No objects found");
        display.display();
    }
#else
    ei_printf("Predictions:\r\n");
    for (uint16_t i = 0; i < EI_CLASSIFIER_LABEL_COUNT; i++) {
        ei_printf(" %s: ",
ei_classifier_inferencing_categories[i]);
        ei_printf("%.5f\r\n", result.classification[i].value);
    }
#endif
#if EI_CLASSIFIER_HAS_ANOMALY
    ei_printf("Anomaly prediction: %.3f\r\n", result.anomaly);
#endif
#if EI_CLASSIFIER_HAS_VISUAL_ANOMALY
    ei_printf("Visual anomalies:\r\n");
    for (uint32_t i = 0; i < result.visual_ad_count; i++) {
        ei_impulse_result_bounding_box_t bb =
result.visual_ad_grid_cells[i];
        if (bb.value == 0) {
            continue;
        }
        ei_printf(" %s (%f) [ x: %u, y: %u, width: %u, height: %u
]\r\n",
                bb.label,
                bb.value,
                bb.x,
                bb.y,
                bb.width,
                bb.height);
    }
#endif
    free(snapshot_buf);
}
@brief
* @retval false if initialisation failed
bool ei_camera_init(void) {
    if (is_initialised) return true;
#if defined(CAMERA_MODEL_ESP_EYE)
    pinMode(13, INPUT_PULLUP);
#endif
}

```

```

    pinMode(14, INPUT_PULLUP);
#endif
    esp_err_t err = esp_camera_init(&camera_config);
    if (err != ESP_OK) {
        Serial.printf("Camera init failed with error 0x%x\n", err);
        return false;
    }
    sensor_t * s = esp_camera_sensor_get();
    if (s->id.PID == OV3660_PID) {
        s->set_vflip(s, 1); // flip it back
        s->set_brightness(s, 1); // up the brightness just a bit
        s->set_saturation(s, 0); // lower the saturation
    }
#if defined(CAMERA_MODEL_M5STACK_WIDE)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
#elif defined(CAMERA_MODEL_ESP_EYE)
    s->set_vflip(s, 1);
    s->set_hmirror(s, 1);
    s->set_awb_gain(s, 1);
#endif
    is_initialised = true;
    return true;
}

@brief
void ei_camera_deinit(void) {
    //deinitialize the camera
    esp_err_t err = esp_camera_deinit();
    if (err != ESP_OK)
    {
        ei_printf("Camera deinit failed\n");
        return;
    }
    is_initialised = false;
    return;
}

@brief      Capture, rescale and crop image
@param[in]  img_width      width of output image
@param[in]  img_height     height of output image

```

```

    @param[in] out_buf          * @retval    bool
ei_camera_capture(uint32_t img_width, uint32_t img_height, uint8_t
*out_buf) {
    bool do_resize = false;
    if (!is_initialised) {
        ei_printf("ERR: Camera is not initialized\r\n");
        return false;
    }
    camera_fb_t *fb = esp_camera_fb_get();
    if (!fb) {
        ei_printf("Camera capture failed\n");
        return false;
    }
    bool converted = fmt2rgb888(fb->buf, fb->len, PIXFORMAT_JPEG,
snapshot_buf);
    esp_camera_fb_return(fb);
    if(!converted){
        ei_printf("Conversion failed\n");
        return false;
    }
    if ((img_width != EI_CAMERA_RAW_FRAME_BUFFER_COLS)
        || (img_height != EI_CAMERA_RAW_FRAME_BUFFER_ROWS)) {
        do_resize = true;
    }
    if (do_resize) {
        ei::image::processing::crop_and_interpolate_rgb888(
            out_buf,
            EI_CAMERA_RAW_FRAME_BUFFER_COLS,
            EI_CAMERA_RAW_FRAME_BUFFER_ROWS,
            out_buf,
            img_width,
            img_height);
    }
    return true;
}
static int ei_camera_get_data(size_t offset, size_t length, float
*out_ptr)
{
    size_t pixel_ix = offset * 3;
    size_t pixels_left = length;
    size_t out_ptr_ix = 0;

```

```
    while (pixels_left != 0) {
        out_ptr[out_ptr_ix] = (snapshot_buf[pixel_ix + 2] << 16) +
(snapshot_buf[pixel_ix + 1] << 8) + snapshot_buf[pixel_ix];
        out_ptr_ix++;
        pixel_ix+=3;
        pixels_left--;
    }
    return 0;
}
#if !defined(EI_CLASSIFIER_SENSOR) || EI_CLASSIFIER_SENSOR !=
EI_CLASSIFIER_SENSOR_CAMERA
#error "Invalid model for current sensor"
#endif
```